ROLE OF PARAMETRIC PROGRAMMING IN NETWORK FLOW PROBLEMS

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree of
DOCTOR OF PHILOSOPHY

By

R. K. AHUJA

to the
INDUSTRIAL AND MANAGEMENT ENGINEERING PROGRAMME
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
JULY, 1981

To

One friend - for his valuable guidance at every step of my life and at every level of my being.

CERTIFICATE

This is to certify that the work embodied in the thesis 'ROLE OF PARAMETRIC PROGRAMMING IN NETWORK FLOW PROBLEMS' by R.K. Ahuja has been carried out under our supervision and has not been submitted elsewhere for a degree or diploma.

(S.K. Gupta)
Asstt. Professor
Department of Mathematics
Indian Institute of Technology,
Kanpur

(J.L. Batra)
Professor and Head
Industrial and Management Engg.Programme
Indian Institute of Technology, Kanpur

July - 1981

ACKNOWLEDGEMENTS

I am grateful to my thesis supervisors, Dr. J.L. Batra and Dr. S.K. Gupta, for responsible guidance, illuminating discussions and constructive criticisms. Indeed, they are responsible for taking out the best in me.

I am also grateful to Dr. A.K. Mittal, Dr. Kripa Shankar and Dr. S. Sadagopan for making me capable of doing research as well as their encouragement and timely helps throughout this research work.

I am thankful to Mr. S.K. Tewari for excellent typing.

R.K. Ahuja

CONTENTS

SYNOPSIS

Network flow problems have received increasing attention in recent years due to significant advances in implementation technology, computer software and modelling techniques. In this dissertation, we study the role of parametric programming in solving some generalizations of the minimum cost flow problem. It is established that parametric programming can be used to obtain simple and intuitive algorithms for several network problems. In doing so, we preserve some of the essential features of most network flow problems, namely, the basis triangularity.

Since our approach is based on parametric programming for network flow problems, we first study parametric analysis of some network problems to get better insight. We consider the maximum flow problem, the minimum spanning tree problem and the shortest path problem and develop procedures for their parametric analysis. The analysis is based on the linear programming formulation of the network problems. We show how this analysis can be used to solve the minimum ratio spanning tree problem and the minimum ratio path problem in acyclic networks. We also consider capacitated transshipment networks in which the supplies at source nodes, demands at sink nodes and capacities of arcs are linear functions of a single parameter. An algorithm based on the parametric analysis of the transshipment problem is developed to determine all values of the parameter for which a feasible flow exists. When the transshipment network is s-t planar, properties of the dual network are

used to develop a simpler algorithm based on the parametric analysis of the shortest path problem.

We next consider the constrained network capacity expansion problem which is essentially a generalization of the constrained minimum cost flow problem and the network capacity expansion problem. In networks, where capacities of arcs can be increased by incurring some additional cost, the constrained network capacity expansion problem is to minimize a linear objective function with the additional constraint that the cost of flow and capacity expansion does not exceed a prescribed budget D. We suggest a parametric approach to solve this problem. We define the concept of optimum basis structure for the constrained network capacity expansion problem. The optimum basis structure is then used to parametrize D. The algorithm produces an optimum solution for all values of D greater than the prescribed value. Almost all the computations are performed in integers and a near-optimum integer solution is readily available. It is demonstrated that the algorithm can be used to solve a variant of the constrained network capacity expansion problem and a bicriteria network problem. Some special cases of the constrained network capacity problem are also considered and resulting simplifications in the algorithm are pointed out.

We also consider capacity expansion of a capacitated transshipment network where the supplies at source nodes and demands at sink nodes are linear functions of a parameter. An algorithm is suggested to determine the least cost of increasing arc capacities for all values of the parameter for which a feasible flow exists.

The approach for the constrained network capacity expansion problem is extended to develop an algorithm for the minimum cost flow problem with piecewise linear convex cost functions. It is wellknown that this convex cost network flow problem can be transformed to the minimum cost flow problem by introducing additional variables. These additional variables are considered implicitly by defining the concept of optimum basis structure for the convex cost network flow problem. The optimum basis structure is then used to parametrize v, the flow to be transshipped from source to sink. The resulting algorithm implicitly enumerates shortest paths from source to sink and augments flow over these paths. The algorithm is shown to be polynomially bounded.

We next show that the proposed algorithm for the convex cost network flow problem can also be used to solve the following problems:

(i) the time-cost tradeoff problem in a CPM network where time-cost tradeoff functions for various arcs are given by piecewise linear convex functions;

(ii) the maximum flow problem in a network with piecewise linear concave gain functions; and

(iii) optimum allocation of a given budget to increase capacities of various arcs in order to maximize the flow in a network where costs of capacity expansion are given by piecewise linear convex functions.

The algorithm for the convex cost network flow problem has been coded and tested on a number of problems. Computational results indicate that the algorithm can solve large-sized network problems in reasonable amount of time.

CHAPTER I

INTRODUCTION

1.1 *INTRODUCTION*

Network flow problems are well-known for their diverse applications as well as efficient solution procedures. Apart from the obvious transportation and communication applications, a wide variety of problems arising in facility location [13, 23, 76, 77], production planning [25, 32, 91, 95], project management [40, 58], machine loading [15, 29], operations scheduling [82, 87], cash management [44, 88] can be formulated as network flow problems. Recent developments in solution methodology, implementation technology and improved computer software have enabled the network flow problems to be handled more effectively and conveniently.

The minimum cost flow problem is perhaps the most popular network flow problem. Some of the reasons for its popularity are: (i) simplicity and intuitive appeal of the problem as well as its solution procedures; (ii) integer valued optimum solution; (iii) little computer memory requirement; (iv) efficiency of computer codes in solving very large-sized problems; and (v) the ability to formulate several complex problems as a sequence of the minimum cost flow problem.

Practical considerations such as losses and gains associated with flows, additional constraints arising due to budgetary and time

constraints, fixed charges associated with new routes, nonlinear costs which are often encountered in practice etc. have led the researchers to investigate generalizations of the minimum cost flow problem. Such generalizations have been sought with a view to enhance the scope of applicability of the minimum cost flow problem. It is obvious that the structure of the original problem is affected, to a certain extent, by these generalizations. The essential effort has been to consider only those generalizations for which it is possible to retain computational efficiency of the solution procedures.

In this dissertation, we explore the use of parametric programming to exploit special structures of some generalizations of the minimum cost flow problem. Specifically, we consider the following problems:

(i) the minimum cost flow problem with an additional linear constraint;

(ii) the network capacity expansion problems;

(iii) the minimum cost flow problem with piecewise linear convex cost functions;

(iv) the time-cost tradeoff problem in CPM networks; and

(v) the maximum flow problem in networks with piecewise linear concave gain functions.

The constrained minimum cost flow problem and the network capacity expansion problem have been studied by several researchers. We consider these two problems in a unified model, which we term as the constrained network capacity expansion problem. The approach commonly used to solve a constrained minimum cost flow problem is to develop the simplex

adaptation for it. If the aspect of capacity expansion is incorporated in the constrained minimum cost flow problem, the simplex adaptation does not exploit the structure effectively. We show that if parametric programming is used to solve the constrained network capacity expansion problem, the structure can be exploited very effectively. The main advantage of this approach is that the basis triangularity is preserved. Furthermore, a near-optimum integer solution is readily available. Some variations and special cases of the constrained network capacity expansion problems are considered. We also consider capacity expansion of a transshipment network where supplies and demands are linear function of a parameter. An algorithm is outlined to determine the least cost of increasing arc capacities for all values of the parameter for which a feasible flow exists.

The parametric approach for the constrained network capacity expansion problem is extended to develop an efficient algorithm for the convex cost network flow problem. The algorithm implicitly enumerates shortest paths from source to sink and augments flow over these paths. It is observed that this algorithm can be used to obtain the project cost curve of a CPM network where time-cost tradeoff functions for various arcs are piecewise linear convex functions. The algorithm, then, implicitly enumerates a cutset at each iteration and modifies the activity times of all activities belonging to this cutset. We also show that the algorithm for the convex cost network flow problem can be used (i) to obtain the maximum flow in a network with piecewise linear concave gain functions; and (ii) to optimally allocate a

prescribed budget to increase the capacities of various arcs to maximize the flow in a network where cost of capacity expansion is given by piecewise linear convex functions.

Since our approach is essentially based on parametric programming, a better insight into it is gained by studying the parametric analysis of some network problems. Specifically, we consider the maximum flow problem, the minimum spanning tree problem and the shortest path problem and develop efficient solution procedures for their parametric analysis. We show how this analysis can be used to solve some minimum ratio network problems and a feasibility problem arising in capacitated transshipment networks.

1.2 *PRELIMINARIES*

Some notations and well-known concepts of graph theory are used throughout the thesis. For the sake of completeness, they are given below.

A directed graph $G = (N,A)$, consists of a finite set N of elements, called nodes, and a set A of ordered pairs of nodes called arcs. A directed network is a directed graph in which numerical values are attached to the nodes and arcs of the graph. Let $n = |N|$ and $m = |A|$. The two specified nodes s and t are called the source and the sink respectively.

An arc (i,j) has two end points, i and j, and it is said to be incident from node i and incident to node j. Let $I(i)$ and $O(i)$

denote, respectively, the sets of arcs incident to and incident from node i. The <u>degree</u> of a node i is the number of arcs incident to or incident from that node.

A <u>path</u> in $G = (N,A)$ is a sequence $i_1, i_2, \ldots, i_r$ of distinct nodes of N such that either $(i_k, i_{k+1}) \varepsilon A$ or $(i_{k+1}, i_k) \varepsilon A$ for each $k=1,\ldots,r-1$. A <u>directed path</u> is defined similarly, except that $(i_k, i_{k+1}) \varepsilon A$ for each $k=1,\ldots,r-1$. A <u>cycle</u> is a path together with an arc (i_r, i_1) or (i_1, i_r). A <u>directed cycle</u> is a directed path together with the arc (i_r, i_1).

A graph $G' = (N', A')$ is a <u>subgraph</u> of $G = (N,A)$ if $N' \subseteq N$ and $A' \subseteq A$. A graph $G' = (N', A')$ is a <u>spanning subgraph</u> of $G = (N,A)$ if $N' = N$ and $A' \subseteq A$.

Two nodes i and j are said to be <u>connected</u> if there is atleast one path between them. A graph is said to be <u>connected</u> if all pairs of nodes are connected; otherwise it is called <u>disconnected</u>. A set $Q \subseteq A$ such that the graph $G' = (N, A-Q)$ is disconnected and no subset of Q has this property, is called a <u>cocycle</u> of G. A cocycle is a <u>cutset</u> if it disconnects source and sink.

A graph is <u>acyclic</u> if it does not contain any cycle. A <u>tree</u> is a connected acyclic graph. A <u>subtree</u> of a tree T is a subgraph of T as well as a tree. A tree T is said to be a <u>spanning tree</u> of G if T is a spanning subgraph of G. Arcs belonging to a spanning tree T are called <u>tree-arcs</u> and arcs not belonging to T are called <u>nontree-arcs</u>. A spanning tree of $G = (N,A)$ has exactly $(n-1)$ tree-arcs.

In a spanning tree, there is a unique path between any two nodes. Addition of any non-tree arc to a spanning tree creates exactly one cycle. If any arc in this cycle is dropped, the resulting graph is again a spanning tree. If any tree-arc is dropped from a spanning tree, two subtrees are formed. Arcs which have their end points belonging to the different subtrees constitute a cocycle. This cocycle is a cutset if the dropped arc belongs to the unique path in the spanning tree from source to sink. If any arc belonging to this cocycle is added to the subtrees, the resulting graph is again a spanning tree.

1.3 *OUTLINE OF THE THESIS*

In Chapter II, we study the parametric analysis of some network problems. The following problems are considered:

(i) the maximum flow problem;

(ii) the minimum spanning tree problem; and

(iii) the shortest path problem.

In the maximum flow problem, the lower and upper bounds on the flow in each arc are considered to be linear functions of a parameter λ. In the minimum spanning tree and shortest path problems, arc lengths are parametrized. The parametric analysis is based on the linear programming formulations of the problems and the structure embedded in the problems is used to simplify the computations. We show how this analysis can be used to solve (i) the minimum ratio spanning tree problem; and (ii) the minimum ratio path problem in acyclic networks.

We also consider capacitated transshipment networks in which the supplies at source nodes, demands at sink nodes and capacities of arcs are linear functions of the parameter λ. An algorithm based on the parametric analysis of the transshipment problem is developed to determine all values of λ for which a feasible flow exists. A simpler algorithm is outlined for s-t planar networks. Various special cases of this problem have been considered by researchers. Our contribution is in proposing simpler algorithms for more general classes of networks.

In Chapter III, we consider a generalization of the constrained minimum cost flow problem and the network capacity expansion problem, which we term as the Constrained Network Capacity Expansion (CNCE) problem. The mathematical statement of the CNCE problem is as follows:

$$\text{Minimize} \quad Z = \sum_{(i,j) \in A} c_{ij}\, x_{ij} \,, \tag{1.1}$$

subject to

$$\sum_{(j,i) \in I(i)} x_{ji} - \sum_{(i,j) \in O(i)} x_{ij} = \begin{cases} -v, & \text{if } i = s, \\ 0, & \text{if } i \neq s,t, \quad \forall\, i \in N, \\ v, & \text{if } i = t, \end{cases} \tag{1.2}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij}, \quad \forall\, (i,j) \in A, \tag{1.3}$$

$$0 \leq y_{ij} \leq a_{ij} - b_{ij}, \quad \forall\, (i,j) \in A, \tag{1.4}$$

$$\sum_{(i,j) \in A} e_{ij}\, x_{ij} + \sum_{(i,j) \in A} f_{ij}\, y_{ij} \leq D, \tag{1.5}$$

where x_{ij} is the flow and y_{ij} is the capacity expansion of arc (i,j).

Problems related to the constrained minimum cost flow problem have been considered by researchers and their emphasis is on developing primal simplex adaptations [43, 54, 62]. This approach, however, does not appear to be promising for the CNCE problem. The constraints (1.3) are not ordinary upper bound restrictions and may not be considered implicitly in the simplex adaptation. Hence the need for an alternate approach arises.

We suggest a parametric approach to solve the CNCE problem. Properties of the optimum solution of the CNCE problem and extensions of the concepts of bounded variable linear programs are combined to evolve the concept of optimum basis structure for the CNCE problem. The optimum basis structure considers the variables y_{ij} and the constraints (1.3) and (1.4) implicitly, thereby reducing the size of the optimum basis considerably. Furthermore, the constraint (1.5) is considered in a manner which does not affect the triangularity of the basis.

The optimum basis structure is used to parametrize D. Initially, the algorithm obtains an optimum basis structure for a large value of D. The value of D is then decreased continuously and an optimum basis structure is maintained at each step until either the prescribed value of D is reached or infeasibility of the problem is indicated. Each iteration of the algorithm is similar to the primal simplex iteration of the minimum cost flow problem. Almost all the computations are performed in integers and a near-optimum integer solution is readily available.

We show that the CNCE algorithm can be used (i) to solve a variation of the CNCE problem; and (ii) to obtain efficient solutions of a bicriteri network problem. Some special cases of the CNCE problem are also conside and the resulting simplifications in the CNCE algorithm are pointed out.

We also consider the capacity expansion of a capacitated transshipme network where supplies at source nodes and demands at sink nodes are linea functions of the parameter λ. The problem considered is to determine the least cost of increasing arc capacities so that the changing demands are met by the changing supplies resulting from the changes in λ. An algorithm, essentially similar to the CNCE algorithm, is developed to solve this capacity expansion problem for all values of the parameter for which a feasible flow exists.

The ideas developed in Chapter III are further refined and extended in Chapter IV to develop an efficient algorithm for the Convex Cost Network Flow (CCNF) problem.

The CCNF problem is to determine the minimum cost flow in a network when cost of flow over each arc is given by a piecewise linear convex function. The CCNF problem has been studied by several researchers and following approaches have been suggested to handle such problems:

(i) the primal-dual approach [64, 70, 71] ;

(ii) the shortest path approach [51] ; and

(iii) the negative cycle approach [11, 35, 36, 59, 67, 93] .

It is well-known that the CCNF problem can be transformed to the minimum cost flow problem by introducing one variable for each linear

segment [72] . We note that in the optimum solution, all the additional variables can be considered implicitly. This observation allows us to define the optimum basis structure for the CCNF problem. The optimum basis structure is then used to parametrize v, the flow to be transshipped from source to sink. Initially, the algorithm obtains an optimum basis structure for $v = 0$. The value of v is then increased and an optimum basis structure is maintained at every step until either the desired flow is established or infeasibility of the problem is indicated. The resulting algorithm successively augments the flow on the shortest paths from source to sink which are implicitly enumerated by the algorithm. Hence the algorithm essentially belongs to the shortest path approach. The computational complexity of the algorithm is shown be $O(mnv)$.

Hu's algorithm [51] for the CCNF problem is also based on the shortest path approach. His algorithm solves a shortest path problem at each iteration and augments flow over that path. Our algorithm generates these paths without solving shortest path problems and hence is more efficient than Hu's algorithm.

We show that the CCNF algorithm can be used to obtain the project cost curve of a CPM network when time-cost tradeoff functions for various activities are given by piecewise linear convex functions. The algorithm, then, identities a cutset at each iteration and activity times of all the activities belonging to this cutset are changed so that the project duration is decreased using minimum additional cost. A similar algorithm to obtain the project cost curve is suggested by Phillips and Dessouki [78] . The basic difference between the two algorithms is

that the cutsets are obtained by applying a cut search procedure at each iteration in Phillips and Dessouky's algorithm, whereas they are implicitly enumerated by our algorithm. The algorithm of Phillips and Dessouky as well as those of Fulkerson [40] and Kelly [58] are primarily suggested for networks with linear time-cost tradeoff functions but can handle piecewise linear convex functions by introducing one arc for each linear segment. This results in enlarging the network substantially. Our algorithm does not require any additional arc to be introduced.

We also establish a relationship between the CCNF problem and the maximum flow problem with piecewise linear concave gain functions. Modifications in the CCNF algorithm are suggested to solve this problem.

The problem of optimally allocating a given budget to increase the capacities of various arcs to maximize the flow in a network, is considered by several researchers for different capacity expansion costs. We show that when capacity expansion costs are given by piecewise linear convex functions, this problem can be solved by the CCNF algorithm.

To test the computational performance of the CCNF algorithm, a program was written and tested on a number of network problems. Computational results demonstrate the efficiency of CCNF algorithm in solving large-sized problems.

Chapter V contains the concluding remarks.

Since the problems considered in various chapters are quite different, we survey the literature chapterwise.

CHAPTER II

PARAMETRIC ANALYSIS OF SOME NETWORK PROBLEMS

2.1 *INTRODUCTION*

Algorithms for several optimization problems solve the shortest path problem or the maximum flow problem as their subproblems [1, 28, 51, 78] . Generally, data of the subproblems generated in two consecutive iterations is different only for a few arcs. Thus, if instead of solving the subproblems afresh, parametric analysis is used to take care of the changes in data, significant computational savings can be achieved.

Parametric programming for network problems was initiated by Srinivasan and Thompson [85, 86] , who developed an operator theory of parametric programming for the transportation problem. This work was subsequently extended to the generalized transportation problem by Balachandran and Thompson [5,6,7,8] . Some results are also available on the sensitivity analysis of network problems [28,38, 83, 94] .

In this chapter, we develop efficient methods to perform parametric analysis of the following network problems:

(i) the maximum flow problem;

(ii) the minimum spanning tree problem; and

(iii) the shortest path problem.

In the maximum flow problem, the lower and upper bounds on the flow in each arc are considered to be linear functions of the parameter λ. In the minimum spanning tree and shortest path problems, arc lengths are parametrized. We show how this analysis can be used to solve (i) minimum ratio spanning tree problem; and (ii) the minimum ratio path problem for acylic networks.

We also consider multiple-source, multiple-sink directed networks in which the supply at each source node, the demand at each sink node and the capacity of each arc is a linear function of the parameter λ. The problem considered is to determine all the values of λ for which a feasible flow exists. This problem has been considered by several researchers for certain restricted classes of networks. Doulliez and Rao [27] considered single-source, multiple-sink networks where only the demand at each sink node <u>increased</u> linearly with time. In [28], they studied the same problem for s-t planar networks and suggested an algorithm which obtained the optimum solution by solving a sequence of shortest path problems. Minieka [68] extended the problem of Doulliez and Rao [27] by considering networks with multiple-sources and multiple-sinks. In this chapter, we propose simple and efficient algorithms for more general networks in which the capacities of arcs are also considered to be linear functions of the parameter λ.

We consider the linear programming formulations of the network problems to perform the parametric analysis and the structure embedded in the problems is used to simplify the computations. Initially, an

optimum basis for some value of λ is obtained. Optimality criteria of this basis is then used to determine all values of λ for which this basis continues to remain optimum. These values of λ constitute an interval which is known as the <u>characteristic interval</u> associated with that basis. At the end points of the characteristic interval, a simplex iteration or a dual simplex iteration is performed to obtain an alternate optimum basis which may allow further increase or decrease in λ. The feasibility problem cited above is also solved by this approach.

2.2 *PARAMETRIC ANALYSIS OF MAXIMUM FLOW PROBLEM*

In this section, we suggest a method to perform the parametric analysis of the maximum flow problem when the lower and upper bounds on the flow in each arc are linear functions of the parameter λ.

Let four numbers a^o_{ij}, a^*_{ij}, b^o_{ij} and b^*_{ij} be associated with each arc $(i,j) \varepsilon A$. Let $a^o_{ij} + \lambda a^*_{ij}$ and $b^o_{ij} + \lambda b^*_{ij}$ be the lower and upper bounds, respectively, on the flow in each arc $(i,j) \varepsilon A$. We assume that the network contains the **fictitious arc** (t,s) for which $a^o_{ts} = a^*_{ts} = b^*_{ts} = 0$ and $b^o_{ts} = \infty$. Then, for a fixed value of λ, the maximum flow problem can be stated as

$$\text{Minimize} \quad - x_{ts}\,, \tag{2.1}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = 0, \quad \forall\, i \,\varepsilon\, N, \tag{2.2}$$

$$a^o_{ij} + \lambda a^*_{ij} \le x_{ij} \le b^o_{ij} + \lambda b^*_{ij}\,, \quad \forall\, (i,j)\,\varepsilon\, A. \tag{2.3}$$

Panwalkar [75] has studied the parametric analysis of bounded variable linear programs. The approach suggested by us for the parametric analysis of the maximum flow problem is similar to that of Panwalkar [75].

2.2.1 Initial Optimum Basis Structure

A basic feasible solution of the maximum flow problem consists of the basic variables and the nonbasic variables which are at their lower or upper bounds. Let B,L and U represent the sets of arcs corresponding to the basic variables, and the nonbasic variables at their lower and upper bounds respectively. We refer to B as a basis and arcs belonging to B as basic arcs. The triplet (B,L,U) is referred to as a basis structure.

The basis of the maximum flow problem is a spanning tree, which consists of two subtrees joined by the arc (t,s) [22]. Arcs which have their end points belonging to the different subtrees constitute a cutset. In the optimum basic feasible solution, this cutset is the minimum cutset. We assume that a value of λ, say λ_o, is known for which the maximum flow problem is feasible. Let $\tilde{x}_{ij}$ be the optimum flow and (B,L,U) be the optimum basis structure for $\lambda = \lambda_o$.

2.2.2 Characteristic Interval

It is easy to see that if λ is varied, the dual feasibility of the optimum basis structure is not affected, but the primal feasibility

may be violated due to changing lower and upper bounds. Thus, the characteristic interval of (B,L,U) consists of all the values of λ for which this basis structure continues to remain feasible.

The characteristic interval is determined from the following considerations:

(i) the flow in each arc $(i,j) \varepsilon B$ does not violate its lower or upper bound;

(ii) the flow in each arc $(i,j) \varepsilon L$ is kept at its changing lower bound; and

(iii) the flow in each arc $(i,j) \varepsilon U$ is kept at its changing upper bound.

It follows from these considerations that as long as the current basis structure remains feasible, flow in each arc $(i,j) \varepsilon B$ is of the form $\tilde{x}_{ij} + (\lambda - \lambda_o) z_{ij}$. The following theorem derives an expression for z_{ij}.

<u>Theorem 2.1</u>: Let T_i and T_j be the resulting subtrees, containing node i and node j respectively, when any arc $(i,j) \varepsilon B$ is dropped from the basis. Let $\bar{Q}_{ij}$ and $\underline{Q}_{ij}$ be the sets of arcs from T_i to T_j and T_j to T_i respectively. Then

$$z_{ij} = \sum_{(k,\ell) \varepsilon \underline{Q}_{ij} \cap U} b^*_{k\ell} - \sum_{(k,\ell) \varepsilon \bar{Q}_{ij} \cap U} b^*_{k\ell} + \sum_{(k,\ell) \varepsilon \underline{Q}_{ij} \cap L} a^*_{k\ell}$$

$$- \sum_{(k,\ell) \varepsilon \underline{Q}_{ij} \cap L} a^*_{k\ell} \ . \qquad (2.4)$$

Proof : Addition of any arc $(k,\ell) \in L \cup U$ to the basis creates exactly one cycle consisting of basic arcs. Let $\overline{W}_{k\ell}$ and $\underline{W}_{k\ell}$ be the sets of arcs in this cycle along and opposite to the orientation of arc (k,ℓ). The flow in each arc $(k,\ell) \in U$ increases at the rate of $b^*_{k\ell}$ as λ is increased. Keeping in view the flow conservation at all the nodes, it amounts to increasing flow in the cycle at the rate of $b^*_{k\ell}$. If the arc $(i,j) \in \overline{W}_{k\ell}$, then flow over it increases at the rate of $b^*_{k\ell}$; and if $(i,j) \in \underline{W}_{k\ell}$, then flow over it decreases at the rate of $b^*_{k\ell}$. We further note that $(i,j) \in \overline{W}_{k\ell}$ iff $(k,\ell) \in \underline{Q}_{ij}$, and $(i,j) \in \underline{W}_{k\ell}$ iff $(k,\ell) \in \overline{Q}_{ij}$. Thus the first two terms in (2.4) represent the effect of changing upper bounds of all the arcs in U on the flow in arc (i,j). Similarly, the last two terms in (2.4) represent the effect of changing lower bounds of all the arcs in L on the flow in arc (i,j).

An efficient method to determine z_{ij} for each $(i,j) \in B$ is stated below. It is easy to see that the method terminates in $O(m)$ iterations.

Step 1: Set $\pi_i = 0$, $\forall\, i \in N$. Perform the following operations for each $(k,\ell) \in L \cup U$ and go to Step 2.

$\pi_k = \pi_k + a^*_{k\ell}$ and $\pi_\ell = \pi_\ell - a^*_{k\ell}$, if $(k,\ell) \in L$,

$\pi_k = \pi_k + b^*_{k\ell}$ and $\pi_\ell = \pi_\ell - b^*_{k\ell}$, if $(k,\ell) \in U$.

Step 2: In the basis select a node of degree 1, say i, and find the unique basic arc (p,q) such that $p = i$ or $q = i$. If

$p = i$, then set $z_{pq} = -\pi_i$ and modify π_q to $\pi_p + \pi_q$. If $q = i$, then set $z_{pq} = \pi_i$ and modify π_p to $\pi_p + \pi_q$. Delete the arc (p,q) from the basis and repeat this step until no arc remains in the basis.

The justification of this method is as follows:

The flow in each arc $(k,\ell) \varepsilon U$ is kept at its changing upper bound, which increases at the rate of $b^*_{k\ell}$. To do so, additional flow is required at node k and getting available at node ℓ at the rate of $b^*_{k\ell}$. Similarly, on account of any arc $(k,\ell) \varepsilon L$, additional flow is required at node k and getting available at node ℓ at the rate of $a^*_{k\ell}$. Thus π_i, at the end of Step 1, denotes the cumulative requirement of flow rate at node i. The tree structure of the basis is then used to determine unique values of z_{ij} which satisfy the flow requirement of all nodes.

Having determined z_{ij}, the characteristic interval is obtained by using the following inequalities:

$$a^o_{ij} + \lambda a^*_{ij} \leq \tilde{x}_{ij} + (\lambda-\lambda_o)\ z_{ij} \leq b^o_{ij} + \lambda b^*_{ij}, \ \forall (i,j) \varepsilon B. \tag{2.5}$$

which can be restated as

$$\tilde{a}_{ij} + (\lambda-\lambda_o)\ a^*_{ij} \leq \tilde{x}_{ij} + (\lambda-\lambda_o)\ z_{ij} \leq \tilde{b}_{ij} + (\lambda-\lambda_o)\ b^*_{ij}, \ \forall (i,j) \varepsilon B, \tag{2.6}$$

where $\tilde{a}_{ij} = a^o_{ij} + \lambda_o a^*_{ij}$ and $\tilde{b}_{ij} = b^o_{ij} + \lambda_o b^*_{ij}$.

Let us define the numbers $\overline{\Delta\lambda}^1_{ij}$, $\overline{\Delta\lambda}^2_{ij}$, $\underline{\Delta\lambda}^1_{ij}$ and $\underline{\Delta\lambda}^2$ for each $(i,j) \varepsilon B$ as follows:

$$\overline{\Delta\lambda}^1_{ij} = \begin{cases} (\tilde{x}_{ij} - \tilde{a}_{ij})/(a^*_{ij} - z_{ij}), & \text{if } z_{ij} < a^*_{ij}, \\ \infty, & \text{otherwise,} \end{cases} \tag{2.7}$$

$$\overline{\Delta\lambda}^2_{ij} = \begin{cases} (\tilde{b}_{ij} - \tilde{x}_{ij})/(z_{ij} - b^*_{ij}), & \text{if } z_{ij} > b^*_{ij}, \\ \infty, & \text{otherwise,} \end{cases} \tag{2.8}$$

$$\underline{\Delta\lambda}^1_{ij} = \begin{cases} (\tilde{x}_{ij} - \tilde{a}_{ij})/(a^*_{ij} - z_{ij}), & \text{if } z_{ij} > a^*_{ij}, \\ -\infty, & \text{otherwise,} \end{cases} \tag{2.9}$$

$$\underline{\Delta\lambda}^2_{ij} = \begin{cases} (\tilde{b}_{ij} - \tilde{x}_{ij})/(z_{ij} - b^*_{ij}), & \text{if } z_{ij} < b^*_{ij}, \\ -\infty, & \text{otherwise.} \end{cases} \tag{2.10}$$

Using (2.7)-(2.10), (2.6) reduces to

$$\underline{\Delta\lambda}^1_{ij} \leq (\lambda - \lambda_o) \leq \overline{\Delta\lambda}^1_{ij} \text{ and } \underline{\Delta\lambda}^2_{ij} \leq (\lambda - \lambda_o) \leq \overline{\Delta\lambda}^2_{ij},$$

$$\forall (i,j) \varepsilon B. \tag{2.11}$$

Let

$$\overline{\lambda}_B = \lambda_o + \min_{(i,j)\varepsilon B}.\ \{ \overline{\Delta\lambda}^1_{ij}, \overline{\Delta\lambda}^2_{ij} \}, \tag{2.12}$$

and

$$\underline{\lambda}_B = \lambda_o + \max_{(i,j)\varepsilon B}.\ \{ \underline{\Delta\lambda}^1_{ij}, \underline{\Delta\lambda}^2_{ij} \}. \tag{2.13}$$

The conditions in (2.11) now simplify to

$$\underline{\lambda}_B \leq \lambda \leq \overline{\lambda}_B \, . \tag{2.14}$$

Thus ($\underline{\lambda}_B$, $\overline{\lambda}_B$) is the characteristic interval of (B,L,U). For all values of $\lambda \in (\underline{\lambda}_B, \overline{\lambda}_B)$, (B,L,U) is the optimum basis structure and the optimum flow x_{ij} is given by

$$x_{ij} = \begin{cases} \bar{x}_{ij} + (\lambda-\lambda_o)\, z_{ij} & , \forall (i,j) \in B, \\ \bar{a}_{ij} + (\lambda-\lambda_o)\, a^*_{ij} & , \forall (i,j) \in L, \\ \bar{b}_{ij} + (\lambda-\lambda_o)\, b^*_{ij} & , \forall (i,j) \in U. \end{cases} \tag{2.15}$$

If it is required to solve the maximum flow problem for $\lambda > \overline{\lambda}_B$ ($\lambda < \underline{\lambda}_B$), then a dual simplex iteration is performed to obtain an alternate optimum basis structure for $\lambda = \overline{\lambda}_B$ ($\lambda = \underline{\lambda}_B$).

2.2.3 <u>Dual Simplex Iteration</u>

Let $\bar{x}_{ij}$ denote the optimum flow for $\lambda = \overline{\lambda}_B$. At $\lambda = \overline{\lambda}_B$ the flow in an arc $(p,q) \in B$, for which $\overline{\lambda}_B = \lambda_o + \min. \{ \overline{\Delta\lambda}^1{}_{pq}, \overline{\Delta\lambda}^2{}_{pq}\}$, equals its lower or upper bound. If $\overline{\lambda}_B = \lambda_o + \overline{\Delta\lambda}^1{}_{pq}$, then $z_{ij} < a^*_{ij}$ and $\bar{x}_{ij} = a^o_{ij} + \overline{\lambda}_B\, a^*_{ij}$. However, if $\overline{\lambda}_B = \lambda_o + \overline{\Delta\lambda}^2{}_{pq}$, then $z_{ij} > b^*_{ij}$ and $\bar{x}_{ij} = b^o_{ij} + \overline{\lambda}_B\, b^*_{ij}$. If λ is increased further, without changing the basis structure, the condition (2.5) is violated. Thus a dual simplex iteration is performed to obtain an alternate optimum basis structure for $\lambda = \overline{\lambda}_B$ which may allow further increase in the value of λ.

The dual simplex iteration is performed by dropping the arc (p,q) from the basis and selecting a nonbasic arc to enter the basis. When the arc (p,q) is dropped from the basis, two subtrees are formed. Let T_p and T_q be the resulting subtrees containing node p and node q respectively. All the arcs which have their one end point in T_p and another in T_q constitute a cocycle Q_{pq}. Define the orientation of the cocycle Q_{pq} along (p,q) if (p,q) leaves the basis at its upper bound, and opposite to (p,q) if it leaves the basis at its lower bound. Let $\bar{Q}_{pq}$ and $\underline{Q}_{pq}$ be the sets of arcs in the cocycle along and opposite to its orientation respectively.

Choice of the entering arc depends upon the entries in the associated simplex tableau in the row corresponding to x_{pq}, and the relative cost coefficients of the nonbasic variables. We now state two known results, which may be found in [10, 22, 27].

<u>Property 2.1</u> : A nonbasic variable $x_{k\ell}$ has an entry -1 in the row corresponding to x_{pq} if and only if $(k,\ell) \in E_{pq}$, where $E_{pq} = (\bar{Q}_{pq} \cap L) \cup (\underline{Q}_{pq} \cap U)$.

<u>Property 2.2</u> : Let $(z_{k\ell}-c_{k\ell})$ denote the relative cost coefficient of the variable $x_{k\ell}$. Then

$$(z_{k\ell}-c_{k\ell}) = \begin{cases} 0 \ , & \forall (k,\ell) \notin Q_{ts} \ , \\ -1 \ , & \forall (k,\ell) \in Q_{ts} \ , \end{cases} \tag{2.16}$$

where Q_{ts} is the set of arcs in the minimum cutset.

Let us define the sets $\bar{E}_{pq}$ and $\underline{E}_{pq}$ as follows:

$$\bar{E}_{pq} = \{(k,\ell) \in E_{pq} : (z_{k\ell} - c_{k\ell}) = 0\} , \tag{2.17}$$

$$\underline{E}_{pq} = \{(k,\ell) \in E_{pq} : (z_{k\ell} - c_{k\ell}) = -1\} . \tag{2.18}$$

If E_{pq} is empty, then all the entries in the row corresponding to x_{pq} are nonnegative. This indicates that the dual simplex iteration can not be performed and the maximum flow problem is infeasible for $\lambda > \bar{\lambda}_B$. If E_{pq} is not empty, then any arc belonging to $\bar{E}_{pq}$ can enter the basis. It is easy to see that the basis changes by the dual simplex iteration but the minimum cutset remains unchanged. However, if $\bar{E}_{pq}$ is empty, then any arc belonging to $\underline{E}_{pq}$ can enter the basis. In this case, minimum cutset also changes. The same procedure can be repeated to determine optimum solutions of the maximum flow problem as λ increases further.

To perform the parametric analysis for $\lambda < \underline{\lambda}_B$, an arc $(p,q) \varepsilon B$ satisfying $\underline{\lambda}_B = \lambda_o + \max. \{\Delta\underline{\lambda}^1_{pq}, \Delta\underline{\lambda}^2_{pq}\}$ leaves the basis. The arc (p,q) leaves the basis at its lower bound if $\lambda_B = \lambda_o + \underline{\Delta\lambda}^1_{pq}$ and at its upper bound if $\underline{\lambda}_B = \lambda_o + \underline{\Delta\lambda}^2_{pq}$. The entering arc is selected in the same manner as it was selected for $\lambda = \bar{\lambda}_B$.

2.2.4 Numerical Example

We now solve a numerical example to illustrate various steps of the parametric analysis of the maximum flow problem. The network is shown in Fig. 2.1. For the sake of simplicity, the lower bound on the flow in each arc is assumed to be zero. Nodes 1 and 6 are the

source and sink respectively. The parametric analysis is performed for all feasible values of $\lambda \geq 0$.

The steps of the parametric analysis are tabulated in Table 2.1. In the table, the symbol $\downarrow$ indicates the basic arc leaving the basis and the symbol $\uparrow$ indicates the nonbasic arc entering the basis. Since the parametric analysis is performed for increasing values of λ, the values of $\underline{\Delta\lambda}^{1}_{ij}$ and $\underline{\Delta\lambda}^{2}_{ij}$ are not required. In the first two iterations, basis as well as the minimum cutsets change. In the third iteration, only the basis changes and the minimum cutset remains unchanged. The basis obtained in the fourth iteration is optimum for all values of $\lambda \geq 4$. The basis in various iterations are shown in Fig. 2.2.

2.2.5 A Special Case: Planar Networks

A network is planar if all the arcs connecting the nodes can be drawn in the plane such that no two arcs cross each other. A network is s-t planar if by adding an arc from node t to node s, the resulting network remains planar. Associated with each s-t planar network is a unique dual network, which can be constructed by the method described in [26]. We denote the dual of the planar network $G = (N,A)$ by $G^* = (N^*, A^*)$.

In $G^* = (N^*, A^*)$, there is a source node as well as a sink node. Let s^* and t^* denote the source and sink in G^* respectively. Further, there is one to one correspondence between arcs in A and A^*. It is wellknown [37] that when the lower bound on the flow in each arc is zero,

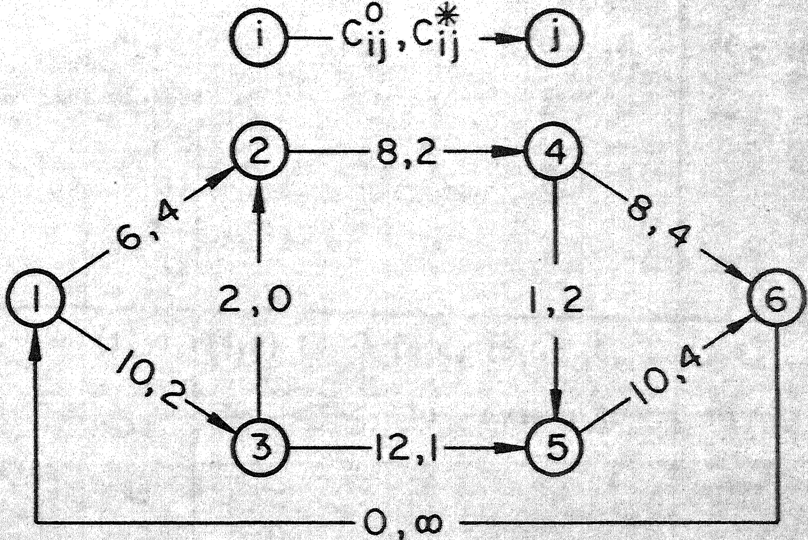

Fig. 2.1 Numerical example for the parametric analysis

Table 2.1 : Solution of the network flow problem

Iteration Number	$\underline{\lambda}_B$	$\overline{\lambda}_B$	Max. Flow		Basic Arcs					Nonbasic Arcs			
				(i,j)	(2,4)	(3,5)	(4,6)	(5,6)	(6,1)	(1,2)	(1,3)	(3,2)	(4,5)
				x_{ij}	6	10	6	10	16	6	10	0	0
1.	0	1	$16+6\lambda$	z_{ij}	4	2	4	2	6	–	–	–	–
				$\overline{\Delta\lambda}^1_{ij}$	∞	∞	∞	∞	∞	–	–	–	–
				$\overline{\Delta\lambda}^2_{ij}$	1	2	∞	∞	∞	–	–	–	–
					↓					↑			
				(i,j)	(1,2)	(3,5)	(4,6)	(5,6)	(6,1)	(2,4)	(1,3)	(3,2)	(4,5)
				x_{ij}	10	12	10	12	22	10	12	0	0
2.	1	2	$22+4\lambda$	z_{ij}	2	2	2	2	4	–	–	–	–
				$\overline{\Delta\lambda}^1_{ij}$	∞	∞	∞	∞	∞	–	–	–	–
				$\overline{\Delta\lambda}^2_{ij}$	∞	1	∞	∞	∞	–	–	–	–
						↓						↑	
				(i,j)	(1,2)	(3,2)	(4,6)	(5,6)	(6,1)	(2,4)	(1,3)	(3,5)	(4,5)
				x_{ij}	12	0	12	14	26	12	14	14	0
3.	2	4	$26+3\lambda$	z_{ij}	1	1	2	1	3	–	–	–	–
				$\overline{\Delta\lambda}^1_{ij}$	∞	∞	∞	∞	∞	–	–	–	–
				$\overline{\Delta\lambda}^2_{ij}$	∞	2	∞	∞	∞	–	–	–	–
						↓					↑		
				(i,j)	(1,2)	(1,3)	(4,6)	(5,6)	(6,1)	(2,4)	(3,2)	(3,5)	(4,5)
				x_{ij}	14	18	16	16	32	16	2	16	0
4.	4	∞	$32+3\lambda$	z_{ij}	2	1	2	1	3	–	–	–	–
				$\overline{\Delta\lambda}^1_{ij}$	∞	∞	∞	∞	∞	–	–	–	–
				$\overline{\Delta\lambda}^2_{ij}$	∞	∞	∞	∞	∞	–	–	–	–

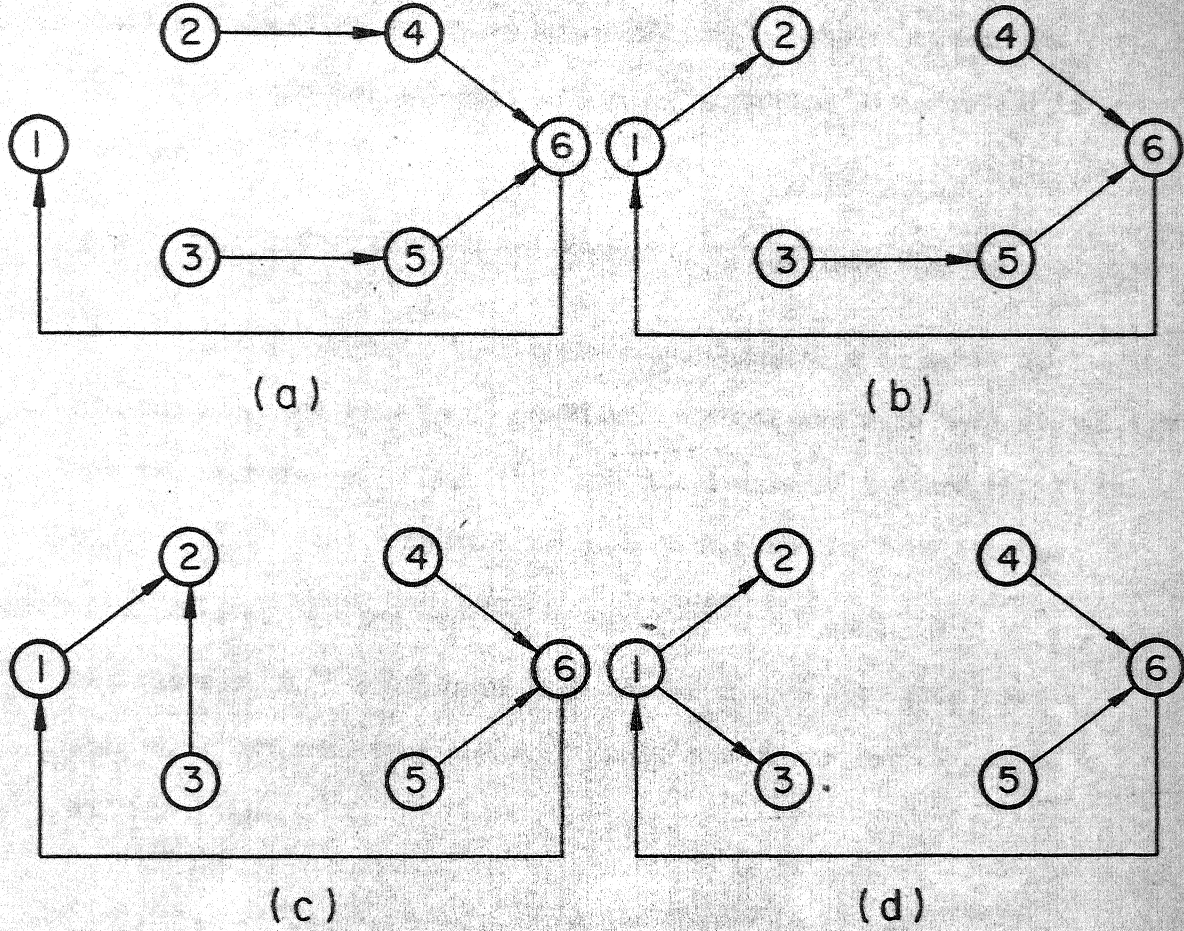

Fig. 2.2 Basis in various iterations

the minimum cutset in G corresponds to the shortest path from s^* to t^* in G^*. In such cases, parametric analysis of the maximum flow problem reduces to the parametric analysis of the shortest path problem, for which a simpler and more efficient algorithm is described in Section 2.4.

2.3 *PARAMETRIC ANALYSIS OF MINIMUM SPANNING TREE PROBLEM*

In this section, we consider the parametric analysis of the minimum spanning tree (MST) problem. Associated with each arc $(i,j) \varepsilon A$ are two numbers c^o_{ij} and c^*_{ij} and the length of the arc (i,j) is $c^o_{ij} + \lambda c^*_{ij}$. The direction of arcs is ignored in this section.

Let T be the minimum spanning tree for some value of λ. In T, let the set $P_{k\ell}$ consist of arcs in the unique path from node k to node ℓ. The necessary and sufficient conditions for T to be a MST [38] are:

$$c^o_{k\ell} + \lambda c^*_{k\ell} \geq c^o_{ij} + \lambda c^*_{ij}, \quad \forall (i,j) \varepsilon P_{k\ell}, \quad \forall (k,\ell) \varepsilon \hat{T}, \tag{2.19}$$

where $\hat{T} = A-T$.

Define the numbers $\bar{\lambda}_{k\ell}$ and $\underline{\lambda}_{k\ell}$ for each $(k,\ell) \varepsilon \hat{T}$ as follows:

$$\bar{\lambda}_{k\ell} = \begin{cases} \min\limits_{(i,j) \varepsilon \bar{E}_{k\ell}} \{ (c^o_{k\ell} - c^o_{ij})/(c^*_{ij} - c^*_{k\ell}) \}, \\ \\ \infty, \quad \text{if } \bar{E}_{k\ell} \text{ is empty,} \end{cases} \tag{2.20}$$

$$\underline{\lambda}_{k\ell} = \begin{cases} \max\limits_{(i,j)\,\varepsilon\,\underline{E}_{k\ell}} \{(c^o_{k\ell}-c^o_{ij})/(c^*_{ij}-c^*_{k\ell})\} , \\ -\infty \quad , \text{ if } \underline{E}_{k\ell} \text{ is empty,} \end{cases} \tag{2.21}$$

where $\bar{E}_{k\ell} = \{(i,j) \;\varepsilon\; P_{k\ell}\colon c^*_{ij} > c^*_{k\ell}\}$ and $\underline{E}_{k\ell} = \{(i,j) \;\varepsilon\; P_{k\ell}\colon c^*_{ij} < c^*_{k\ell}\}$.

Using (2.20) and (2.21), (2.19) reduces to

$$\underline{\lambda}_{k\ell} \leq \lambda \leq \bar{\lambda}_{k\ell} \, , \; \forall \, (k,\ell) \;\varepsilon\; \hat{T}, \tag{2.22}$$

which further simplifies to

$$\underline{\lambda}_T \leq \lambda \leq \bar{\lambda}_T \, , \tag{2.23}$$

where $\bar{\lambda}_T = \min\limits_{(k,\ell)\,\varepsilon\,\hat{T}} \{\bar{\lambda}_{k\ell}\}$ and $\underline{\lambda}_T = \max\limits_{(k,\ell)\,\varepsilon\,\hat{T}} \{\underline{\lambda}_{k\ell}\}$.

Thus $(\underline{\lambda}_T, \bar{\lambda}_T)$ is the characteristic interval for T. For all values of $\lambda \;\varepsilon\; (\underline{\lambda}_T, \bar{\lambda}_T)$, T is a MST and its length is $\left(\sum\limits_{(i,j)\,\varepsilon\,T} c^o_{ij} + \lambda \sum\limits_{(i,j)\,\varepsilon\,T} c^*_{ij}\right)$.

To perform the parametric analysis for $\lambda > \bar{\lambda}_T$, identify an arc $(u,v) \;\varepsilon\; \hat{T}$ for which $\bar{\lambda}_{uv} = \bar{\lambda}_T$ and an arc $(p,q) \;\varepsilon\; \bar{E}_{uv}$ for which $(c^o_{pq} - c^o_{uv})/(c^*_{uv} - c^*_{pq}) = \bar{\lambda}_{uv}$. An alternate MST for $\lambda = \bar{\lambda}_T$ is $T' = T \cup \{(u,v)\} - \{(p,q)\}$. In the similar manner, the parametric analysis for T' can be performed.

Similarly, to perform the parametric analysis for $\lambda < \underline{\lambda}_T$, identify an arc $(u,v) \;\varepsilon\; \hat{T}$ for which $\underline{\lambda}_{uv} = \underline{\lambda}_T$ and an arc $(p,q) \;\varepsilon\; \underline{E}_{uv}$ for which $(c^o_{pq}-c^o_{uv})/(c^*_{uv}-c^*_{pq}) = \underline{\lambda}_{uv}$. An alternate MST for $\lambda = \underline{\lambda}_T$ is $T' = T \cup \{(u,v)\} - \{(p,q)\}$.

The characteristic intervals of MST'S obtained in the course of parametric analysis do not overlap. It is also easy to see that once λ leaves the characteristic interval of a MST, that tree never reappears. The total number of MST's is clearly finite. Moreover, an upper bound on the number of MST's can be stated as follows:

Theorem 2.2 : An upper bound on the number of MST's obtained, as λ varies in the interval $-\infty \leq \lambda \leq \infty$, is $m(m-2)/4$ if m is even, and $(m-1)^2/4$ if m is odd.

Proof. We refer to c^*_{ij} as the incremental length of arc (i,j). Let us number the arcs as $a_1, a_2, \ldots, a_m$ in the increasing order of their incremental lengths and consider the parametric analysis for increasing values of λ. It follows from (2.20) that whenever the MST changes, the entering arc has smaller incremental length than the leaving arc. Consider an arc a_i. There are atmost $(i-1)$ arcs which have incremental lengths less than that of a_i and atmost $(m-i)$ arcs which have incremental lengths greater than that of a_i. If a_i is a tree-arc in a MST, then it may become a nontree-arc by being replaced by a nontree-arc of smaller incremental length which can not happen more than $(i-1)$ times. Similarly, if a_i is a nontree-arc, then it may become a tree-arc in a MST by replacing an arc of larger incremental length and this would not happen more than $(m-i)$ times. Thus a_i may cause MST to change atmost min. $\{i-1, m-i\}$ times. When this value is summed up for all the arcs, we get the desired expressions.

2.4 PARAMETRIC ANALYSIS OF SHORTEST PATH PROBLEM

In this section, we consider the parametric analysis of the shortest path problem. Associated with each arc $(i,j) \varepsilon A$ are numbers c^o_{ij} and c^*_{ij}, and the length of the arc (i,j) is considered to be $c^o_{ij} + \lambda c^*_{ij}$. It is assumed that the network does not contain negative cycles for any value of the parameter λ in the range of the analysis.

The shortest path problem, for a fixed value of λ, can be stated as

$$\text{Minimize} \quad \sum_{(i,j)\,\varepsilon\, A} (c^o_{ij} + \lambda c^*_{ij})\, x_{ij} \,, \tag{2.24}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\varepsilon\, O(i)} x_{ij} = \begin{cases} -(n-1), & \text{if } i=s, \\ & , \forall\, i\,\varepsilon\, N, \\ 1, & \text{otherwise,} \end{cases} \tag{2.25}$$

$$x_{ij} \geq 0, \ \forall\, (i,j)\ \varepsilon\ A. \tag{2.26}$$

It is wellknown [22] that any optimum basis of the shortest path problem is a spanning tree. In this spanning tree, there is a unique directed path from source to every other node and this path is the shortest path from source to that node. Therefore, this spanning tree is known as the <u>tree of shortest paths</u> rooted at source. Let T be the tree of shortest paths for some value of λ. The necessary and sufficient conditions for T to be a tree of shortest paths are:

$$(\pi^o_\ell + \lambda \pi^*_\ell) - (\pi^o_k + \lambda \pi^*_k) \leq c^o_{k\ell} + \lambda c^*_{k\ell}\,, \ \forall\, (k,\ell)\ \varepsilon\ \hat{T}, \tag{2.27}$$

where $\hat{T} = A - T$, and π_j^o and π_j^* are the numbers satisfying

$$\pi_s^o = 0 \quad \text{and} \quad \pi_j^o - \pi_i^o = c_{ij}^o \; , \; \forall \, (i,j) \; \varepsilon \; T, \tag{2.28}$$

$$\pi_s^* = 0 \quad \text{and} \quad \pi_j^* - \pi_i^* = c_{ij}^* \; , \; \forall \, (i,j) \; \varepsilon \; T. \tag{2.29}$$

Let $\overline{\lambda}_{k\ell}$ and $\underline{\lambda}_{k\ell}$ be the numbers defined, for each arc $(k,\ell) \; \varepsilon \; \hat{T}$, as follows:

$$\overline{\lambda}_{k\ell} = \begin{cases} (\pi_k^o - \pi_\ell^o + c_{k\ell}^o)/(\pi_\ell^* - \pi_k^* - c_{k\ell}^*), \text{if } (\pi_\ell^* - \pi_k^* - c_{k\ell}^*) > 0, \\ \infty \; , \text{ otherwise,} \end{cases} \tag{2.30}$$

$$\underline{\lambda}_{k\ell} = \begin{cases} (\pi_k^o - \pi_\ell^o + c_{k\ell}^o)/(\pi_\ell^* - \pi_k^* - c_{k\ell}^*), \text{if } (\pi_\ell^* - \pi_k^* - c_{k\ell}^*) < 0, \\ -\infty \; , \text{ otherwise.} \end{cases} \tag{2.31}$$

Using (2.30) and (2.31), (2.27) reduces to

$$\underline{\lambda}_{k\ell} \leq \lambda \leq \overline{\lambda}_{k\ell} \; , \quad \forall \, (k,\ell) \; \varepsilon \; \hat{T}, \tag{2.32}$$

which further simplifies to

$$\underline{\lambda}_T \leq \lambda \leq \overline{\lambda}_T \; , \tag{2.33}$$

where $\overline{\lambda}_T = \min_{(k,\ell)\, \varepsilon\, \hat{T}} \{\overline{\lambda}_{k\ell}\}$ and $\underline{\lambda}_T = \max_{(k,\ell)\, \varepsilon\, \hat{T}} \{\underline{\lambda}_{k\ell}\}$.

Thus $(\underline{\lambda}_T, \overline{\lambda}_T)$ is the characteristic interval of T. for all values of $\lambda \, \varepsilon \, (\underline{\lambda}_T, \overline{\lambda}_T)$, T is a tree of shortest paths and length of the shortest path from source to any node $j \; \varepsilon \; N$ is $\pi_j^o + \lambda \pi_j^*$.

To perform the parametric analysis for $\lambda > \overline{\lambda}_T$, a simplex iteration is performed to obtain an alternate tree of shortest paths

for $\lambda = \bar{\lambda}_T$. Let $(u,v) \in \hat{T}$ be an arc for which $\bar{\lambda}_{uv} = \bar{\lambda}_T$ and $(p,q) \in T$ be the unique arc for which $q=v$. The tree $T' = T \cup \{(u,v)\} - \{(p,q)\}$ is an alternate tree of shortest paths for $\lambda = \bar{\lambda}_T$. Repeating the same procedure, subsequent tree of shortest paths can be obtained as λ increases further.

Similarly, to perform the parametric analysis for $\lambda < \underline{\lambda}_T$, select an arc $(u,v) \in \hat{T}$ for which $\underline{\lambda}_{uv} = \underline{\lambda}_T$ and the unique arc $(p,q) \in T$ such that $q = v$. The tree $T' = T \cup \{(u,v)\} - \{(p,q)\}$ is an alternate tree of shortest paths for $\lambda = \underline{\lambda}_T$.

2.5 *PARAMETRIC NETWORK FEASIBILITY PROBLEM*

In this section, we consider multiple-source, multiple-sink directed networks in which supplies at source nodes, demands at sink nodes and capacities of arcs are linear functions of a parameter λ. An algorithm, based on the parametric analysis of the transshipment problem, is developed to determine all the values of λ for which a feasible flow exists. A simpler and more efficient algorithm is suggested for s-t planar networks.

Let $S \subset N$ denote the set of all source nodes and $T \subset N$ denote the set of all sink nodes, where $S \cap T = \phi$. Let the supply of source node $i \in S$, the demand at sink node $i \in T$, and capacity of arc $(i,j) \in A$ be $-r_i^o - \lambda r_i^*$, $r_i^o + \lambda r_i^*$ and $b_{ij}^o + \lambda b_{ij}^*$ respectively. Here r_i^o, r_i^*, b_{ij}^o and b_{ij}^* are all fixed and λ is a parameter which is permitted to vary. If for each $i \in N - S \cup T$ we put $r_i^o = r_i^* = 0$, then without loss of generality we can assume that $\sum_{i \in N} r_i^o = \sum_{i \in N} r_i^* = 0$.

Let λ_{min} and λ_{max} denote the smallest and largest value of λ for which a feasible flow of the transshipment network exists. The problem to determine λ_{max} can be formulated as the following linear program:

$$\text{Maximize} \quad \lambda , \tag{2.34}$$

subject to

$$\sum_{(j,i) \varepsilon I(i)} x_{ji} - \sum_{(i,j) \varepsilon O(i)} x_{ij} = r_i^o + \lambda r_i^*, \ \forall \ i \ \varepsilon \ N, \tag{2.35}$$

$$0 \leq x_{ij} \leq b_{ij}^o + \lambda b_{ij}^*, \ \forall \ (i,j) \ \varepsilon \ A. \tag{2.36}$$

We refer to the problem (2.34)-(2.36) as the Parametric Network Feasibility (PNF) problem. Although, it is assumed in the PNF problem that the lower bound on the flow in each arc is zero, it is by no means restrictive. Problems with nonzero or parametric lower bound on the flow in each arc can be converted to the PNF problem by a simple transformation of variables. It should further be noted that the problem to determine λ_{min} can be solved by solving the PNF problem in which b_i^* is replaced by $-b_i^*$ for each $i \ \varepsilon \ N$ and c_{ij}^* is replaced by $-c_{ij}^*$ for each $(i,j) \ \varepsilon \ A$.

In principle, the PNF problem can be solved as a linear program by considering λ as one of the variables. But in doing so, the structure of the constraint matrix is significantly affected. We, therefore, consider λ as a parameter and retain the structure of the problem. We then use the concepts of parametric linear programming to determine the range of feasible values of λ.

2.5.1 Development of the Algorithm

The PNF problem is a bounded variable linear program. In a basic feasible solution of the PNF problem, let B,L and U represent the sets of arcs corresponding to the basic variables and the nonbasic variables which are at their lower and upper bounds respectively. We refer to B as the <u>basis</u> and the triplet (B,L,U) as the <u>basis structure</u>. It is known that the basis of the PNF problem is a spanning tree.

We assume that a value of λ, say λ_o, is known for which the PNF problem is feasible. A basic feasible solution for $\lambda = \lambda_o$ can be obtained by the method described in [10]. In this solution, let $\tilde{x}_{ij}$ represent the flow on arc $(i,j) \in A$ and (B,L,U) be the associated basis structure. If λ is increased, the feasibility of this basis structure may be violated because of changing supply-demand requirements and the changing arc capacities. We first determine the largest value of λ upto which the current basis structure continues to remain feasible. This largest value of λ, denoted by $\bar{\lambda}$, is determined from the following considerations:

(i) The flow in each arc $(i,j) \in B$ remains within its lower and upper bounds;

(ii) the flow in each arc $(i,j) \in L$ is kept at zero value; and

(iii) the flow in each arc $(i,j) \in U$ is kept at its changing upper bound.

It is easy to see that as long as the current basis structure remains feasible, the flow in each arc $(i,j) \varepsilon B$ is of the form $\tilde{x}_{ij} + (\lambda - \lambda_o)\, z_{ij}$. The following theorem derives the expressions for z_{ij}.

<u>Theorem 2.3</u> : Let T_i and T_j be the resulting subtrees, containing node i and node j respectively, when any arc $(i,j) \varepsilon B$ is dropped from the basis. Let $\bar{Q}_{ij}$ and $\underline{Q}_{ij}$ be the sets of arcs from T_i to T_j and T_j to T_i respectively. Then

$$z_{ij} = \sum_{k \varepsilon T_j} r^*_k + \sum_{(k,\ell) \varepsilon \underline{Q}_{ij} \cap U} b^*_{k\ell} - \sum_{(k,\ell) \varepsilon \bar{Q}_{ij} \cap U} b^*_{k\ell}. \qquad (2.37)$$

<u>Proof</u> : The first term in (2.37), i.e., $\sum_{k \varepsilon T_j} r^*_k$, represents the effect of changing supply-demand requirements on the flow in the arc (i,j). It is the rate at which the net demand of nodes in T_j increases with λ. Since $\sum_{k \varepsilon T_i} r^*_k + \sum_{k \varepsilon T_j} r^*_k = 0$, it is also the rate at which the net supply of nodes in T_i increases with λ. Since (i,j) is the only basic arc connecting T_i to T_j, the flow over it must change at the rate of $\sum_{k \varepsilon T_j} r^*_k$ in order to satisfy the changing demands by the changing supplies.

The last two terms in (2.37) indicate the effect of changing arc capacities on the flow in arc (i,j). Proof of Theorem 2.1 includes the justification for these terms.

The following is the procedure to determine z_{ij}. The justification of this procedure is similar to the one used for computing z_{ij} in the maximum flow problem.

Step 1. Set $\pi_i = r_i^*$, $\forall\, i \in N$. Perform the following operations for each $(k,\ell) \in U$ and go to Step 2:

$$\pi_k = \pi_k + b^*_{k\ell} \quad \text{and} \quad \pi_\ell = \pi_\ell - b^*_{k\ell}.$$

Step 2. In the basis select a node of degree 1, say i, and find the unique basic arc (p,q) such that p=i or q=i. If p=i, then set $z_{pq} = -\pi_i$ and modify π_q to $\pi_p + \pi_q$. If q=i, then set $z_{pq} = \pi_i$ and modify π_p to $\pi_p + \pi_q$. Delete the arc (p,q) from the basis and repeat this step until no arc remains in the basis.

Having determined z_{ij}, $\bar{\lambda}$ is determined from the following inequalities:

$$0 \leq \tilde{x}_{ij} + (\lambda-\lambda_o)\, z_{ij} \leq \tilde{b}_{ij} + (\lambda-\lambda_o)\, b^*_{ij},\ \forall\,(i,j) \in B, \tag{2.38}$$

where $\tilde{b}_{ij} = b^o_{ij} + \lambda_o\, b^*_{ij}$

Define the numbers $\overline{\Delta\lambda}^1_{ij}$, $\overline{\Delta\lambda}^2_{ij}$, $\underline{\Delta\lambda}^1_{ij}$ and $\underline{\Delta\lambda}^2_{ij}$ for each $(i,j) \in B$ as follows:

$$\overline{\Delta\lambda}^1_{ij} = \begin{cases} -\tilde{x}_{ij}/z_{ij}, & \text{if } z_{ij} < 0, \\ \infty, & \text{otherwise}, \end{cases} \tag{2.39}$$

$$\overline{\Delta\lambda}^2_{ij} = \begin{cases} (\tilde{b}_{ij}-\tilde{x}_{ij})/(z_{ij}-b^*_{ij}), & \text{if } z_{ij} > b^*_{ij}, \\ \infty, & \text{otherwise}, \end{cases} \tag{2.40}$$

$$\underline{\Delta\lambda}^1_{ij} = \begin{cases} -\tilde{x}_{ij}/z_{ij}, & \text{if } z_{ij} > 0, \\ -\infty, & \text{otherwise}, \end{cases} \tag{2.41}$$

$$\underline{\Delta\lambda}^2_{ij} = \begin{cases} (\tilde{b}_{ij}-\tilde{x}_{ij})/(z_{ij}-b^*_{ij}), & \text{if } z_{ij} < b^*_{ij} , \\ -\infty, & \text{otherwise}. \end{cases} \tag{2.42}$$

Using (2.39)-(2.42), (2.38) reduces to

$$\underline{\Delta\lambda}^1_{ij} \leq (\lambda-\lambda_o) \leq \overline{\Delta\lambda}^1_{ij} \quad \text{and} \quad \underline{\Delta\lambda}^2_{ij} \leq (\lambda-\lambda_o) \leq \overline{\Delta\lambda}^2_{ij}, \ \forall\,(i,j) \in B. \tag{2.43}$$

It is obvious from (2.43) that

$$\bar{\lambda} = \lambda_o + \min_{(i,j)\in B}. \{\Delta\lambda^1_{ij} \ , \ \overline{\Delta\lambda}^2_{ij}\} . \tag{2.44}$$

A feasible flow $\bar{x}_{ij}$ for $\lambda = \bar{\lambda}$ is as follows:

$$\bar{x}_{ij} = \begin{cases} \tilde{x}_{ij} + (\bar{\lambda}-\lambda_o)\, z_{ij} , & \forall\,(i,j) \in B , \\ 0 , & \forall\,(i,j) \in L , \\ \tilde{b}_{ij} + (\bar{\lambda}-\lambda_o)\, b^*_{ij} , & \forall\,(i,j) \in U . \end{cases} \tag{2.45}$$

At $\lambda = \bar{\lambda}$, flow in an arc (p,q) for which $\lambda_o +$ min. $\{\overline{\Delta\lambda}^1_{pq} \ , \ \overline{\Delta\lambda}^2_{pq}\} = \bar{\lambda}$ reaches either zero value or its upper bound. It follows from (2.39) and (2.40) that if flow reaches zero value then $z_{pq} < 0$; and if flow reaches its upper bound, then $z_{pq} > b^*_{pq}$. If λ is increased further without changing the basis structure, flow in arc (p,q) violates its bounds. Thus to find a feasible flow for $\lambda > \bar{\lambda}$, a dual simplex iteration is performed to obtain an alternate basis structure for $\lambda = \bar{\lambda}$.

The dual simplex iteration is performed by dropping the arc (p,q) from the basis and selecting a nonbasic arc to enter the basis.

Dropping the arc (p,q) yields the cocycle Q_{pq}. Define the orientation of the cocycle along (p,q) if (p,q) leaves at its upper bound and opposite to (p,q) if (p,q) leaves at zero value. Let $\bar{Q}_{pq}$ and $\underline{Q}_{pq}$ be the sets of arcs in Q_{pq} along and opposite to its orientation respectively.

The structure of the constraint matrices for the PNF problem and the maximum flow problem are same and, therefore, Property 2.1 remains valid for the PNF problem. The relative cost coefficients for all nonbasic variables are obviously zero. Hence, if the set $E_{pq} = (\bar{Q}_{pq} \cap L) \cup (\underline{Q}_{pq} \cap U)$ is nonempty, then any arc in it can be selected to enter the basis. The same procedure can be repeated for the new basis structure. However, if E_{pq} is empty, then the dual complex iteration can not be performed indicating that $\bar{\lambda} = \lambda_{max}$.

It may be noted that the choice of the arc in E_{pq}, selected to enter the basis, is crucial for the number of iterations required to reach λ_{max}. Several selection rules were tried computationally. Finally, the following selection rule which reduced the number of iterations substantially was selected. It is discussed for two different cases.

<u>Case I</u> : The arc (p,q) leaves the basis at zero value. If any arc $(k,\ell) \in E_{pq}$ enters the basis, then $z_{k\ell}$ in the alternate basis structure is given by

$$z_{k\ell} = \begin{cases} -z_{pq} & , \text{ if } (k,\ell) \in \bar{Q}_{pq} \cap L, \\ z_{pq} + b^*_{k\ell}, & \text{if } (k,\ell) \in \underline{Q}_{pq} \cap U \end{cases} \tag{2.46}$$

Let $\overline{\Delta\lambda}'_{k\ell}$ denote the largest increase in the value of λ, that the arc (k,ℓ) can tolerate without going out of the basis, assuming that all the other basic arcs continue to remain basic. The expression for $\overline{\Delta\lambda}'_{k\ell}$ is as follows:

$$\overline{\Delta\lambda}'_{k\ell} = \begin{cases} \bar{b}_{k\ell}/(-z_{pq}-b^*_{k\ell}), & \text{if } -z_{pq} > b^*_{k\ell}, \\ \infty, & \text{otherwise}, \end{cases} \tag{2.47}$$

where $\bar{b}_{k\ell} = b^o_{k\ell} + \bar{\lambda}\, b^*_{k\ell}$.

<u>Case II</u> : The arc (p,q) leaves the basis at its upper bound. In this case, the expressions for $z_{k\ell}$ and $\overline{\Delta\lambda}'_{k\ell}$ are as follows:

$$z_{k\ell} = \begin{cases} z_{pq} - b^*_{pq} & , \text{ if } (k,\ell) \in \bar{Q}_{pq} \cap L, \\ -z_{pq} + b^*_{pq} + b^*_{k\ell}, & \text{if } (k,\ell) \in \underline{Q}_{pq} \cap U, \end{cases} \tag{2.48}$$

$$\overline{\Delta\lambda}'_{k\ell} = \begin{cases} \bar{b}_{k\ell}/(z_{pq}-b^*_{pq}-b^*_{k\ell}), & \text{if } z_{pq} > b^*_{pq} + b^*_{k\ell}, \\ \infty, & \text{otherwise}. \end{cases} \tag{2.49}$$

The number $\overline{\Delta\lambda}'_{k\ell}$ can be taken as a measure of increase in λ permitted by the arc (k,ℓ) without becoming nonbasic. Hence, if at every iteration an arc in E_{pq} having the largest value of $\overline{\Delta\lambda}'_{k\ell}$ is selected to enter the basis, larger increase in λ per iteration is expected. This conclusion was verified computationally.

2.5.2 Description of the Algorithm

A stepwise description of the algorithm is given below.

Step 1. Obtain a feasible basis structure (B,L,U) for $\lambda = \lambda_o$ by the method described in [10]. Let x_{ij} be the flow on arc $(i,j) \in A$. Set $\bar{\lambda} = \lambda_o$. Let $\bar{b}_{ij} = b^o_{ij} + \bar{\lambda}\, b^*_{ij}$ for each $(i,j) \in A$. Go to Step 2.

Step 2. Compute z_{ij} for each $(i,j) \in B$. Let $\overline{\Delta\lambda}^1_{ij}$ and $\overline{\Delta\lambda}^2_{ij}$ be the numbers defined for each $(i,j) \in B$ as follows:

$$\overline{\Delta\lambda}^1_{ij} = \begin{cases} -x_{ij}/z_{ij} \ , & \text{if } z_{ij} < 0, \\ \infty \ , & \text{otherwise,} \end{cases}$$

$$\overline{\Delta\lambda}^2_{ij} = \begin{cases} (\bar{b}_{ij}-x_{ij})/(z_{ij}-b^*_{ij}) \ , & \text{if } z_{ij} > b^*_{ij} \ , \\ \infty \ , & \text{otherwise.} \end{cases}$$

Let

$$\overline{\Delta\lambda} = \min_{(i,j)\in B} \{ \overline{\Delta\lambda}^1_{ij} \ , \ \overline{\Delta\lambda}^2_{ij} \} .$$

Update $\bar{\lambda}$, $\bar{b}_{ij}$ and x_{ij} as follows:

$$\bar{\lambda} = \bar{\lambda} + \overline{\Delta\lambda} \ ,$$

$$\bar{b}_{ij} = \bar{b}_{ij} + \overline{\Delta\lambda} \quad b^*_{ij} \ , \ \forall (i,j) \in A \ ,$$

$$x_{ij} = \begin{cases} x_{ij} + \overline{\Delta\lambda} \quad z_{ij} \ , & \forall (i,j) \in B \ , \\ 0 \ , & \forall (i,j) \in L \ , \\ \bar{b}_{ij} \ , & \forall (i,j) \in U \ . \end{cases}$$

If $\bar{\lambda} = \infty$, go to Step 4; otherwise identify an arc $(p,q) \varepsilon B$ for which min. $\{\overline{\Delta\lambda}^1_{pq}, \Delta\lambda^2_{pq}\} = \overline{\Delta\lambda}$ and go to Step 3.

Step 3. Dropping the arc (p,q) from the basis yields the cocycle Q_{pq}. Define the orientation of Q_{pq} along (p,q) if $x_{ij} = \bar{b}_{ij}$ and opposite to (p,q) if $x_{ij} = 0$. Let $\bar{Q}_{pq}$ and $\underline{Q}_{pq}$ be the sets of arcs in Q_{pq} along and opposite to its orientation respectively. Let $E_{pq} = (\bar{Q}_{pq} \cap L) \cup (\underline{Q}_{pq} \cap U)$. If E_{pq} is empty, go to Step 4; otherwise define a number $\overline{\Delta\lambda}'_{k\ell}$ for each $(k,\ell) \varepsilon E_{pq}$ as follows:

$$\overline{\Delta\lambda}'_{k\ell} = \begin{cases} \bar{b}_{k\ell}/(-z_{pq}-b^*_{k\ell}), \text{ if } x_{pq} = 0 \text{ and } -z_{pq} > b^*_{k\ell}, \\ \bar{b}_{k\ell}/(z_{pq}-b^*_{pq}-b^*_{k\ell}), \text{ if } x_{pq} = \bar{b}_{pq} \text{ and } z_{pq} > b^*_{pq} + b^*_{k\ell} \\ \infty, \text{ otherwise.} \end{cases}$$

Identify an arc $(u,v) \varepsilon E_{pq}$ for which $\overline{\Delta\lambda}'_{uv} = \max\limits_{(k,\ell) \varepsilon E_{pq}} \{\overline{\Delta\lambda}'_{k\ell}\}$. The arc (u,v) enters the basis. Update the basis structure and go to Step 2.

Step 4: Set $\lambda_{max} = \bar{\lambda}$ and STOP.

2.5.3 A Special Case : Planar Networks

A single-source, single-sink network can always be associated with every multiple-source, multiple-sink network in the manner described as follows: The network $G = (N,A)$ is augmented by a super-source node $\bar{s}$ and a super-sink node $\bar{t}$. An arc $(\bar{s},i)$ is introduced for each $i \varepsilon S$ with $b^o_{\bar{s}i} = -r^o_i$ and $b^*_{\bar{s}i} = -r^*_i$. Similarly, an arc $(i,\bar{t})$ is introduced for each $i \varepsilon T$ with $b^o_{i\bar{t}} = r^o_i$ and $b^*_{i\bar{t}} = r^*_i$. Let the augmented network be denoted by $\bar{G} = (\bar{N},\bar{A})$.

Let us assume that $\bar{G}$ is s–t planar. It can be easily seen that $\bar{G}$ is s–t planar if and only if G is s–t planar for every combination of source–sink pair. The dual network of $\bar{G} = (\bar{N}, \bar{A})$ can be obtained by the method described in [26]. Let $G^* = (N^*, A^*)$ denote the dual network. Let s^* and t^* be the source and sink in G^*. Since there is one to one correspondence between arcs in $\bar{G}$ and G^*, let (i^*, j^*) denote the arc in G^* corresponding to the arc (i,j) in $\bar{G}$. Let $\bar{M} = \{(i,\bar{t}) \in \bar{A}: i \in T\}$ and $M^* = \{(i^*,j^*) \in A^*: (i,j) \in \bar{M}\}$.

It is easy to see that the PNF problem is feasible for some λ if and only if flow in arcs belonging to $\bar{M}$ equals their respective upper bound. This implies that the PNF problem is feasible for some λ if and only if $\bar{M}$ is the minimum cutset with arc capacities as $b^o_{ij} + \lambda b^*_{ij}$. Since the minimum cutset in $\bar{G}$ corresponds to the shortest path from s^* to t^* in G^*, it follows that the PNF problem is feasible for some λ if and only if M^* is the shortest path from s^* to t^* in G^* with arc lengths as $b^o_{ij} + \lambda b^*_{ij}$.

The PNF algorithm for s–t planar networks is now obvious. The parametric analysis of the shortest path from s^* to t^* in G^* is performed for increasing values of λ with arc lengths as $b^o_{ij} + \lambda b^*_{ij}$. The highest value of λ, upto which M^* continues to remain the shortest path from s^* to t^*, is the value of λ_{max}.

2.6 *MINIMUM RATIO NETWORK PROBLEMS*

Parametric programming is used extensively to solve fractional programming problems [24, 55]. Following the same approach, we show

that the parametric analysis of network problems can be used to solve (i) the minimum ratio spanning tree problem; and (ii) the minimum ratio shortest path problem for acyclic networks.

2.6.1 Minimum Ratio Spanning Tree Problem

Let two numbers, c_{ij} and d_{ij}, be associated with each arc $(i,j) \varepsilon A$. The Minimum Ratio Spanning Tree (MRST) problem is to determine a tree T^* for which the ratio

$$\sum_{(i,j)\,\varepsilon\, T^*} c_{ij} \Big/ \sum_{(i,j)\,\varepsilon\, T^*} d_{ij} , \tag{2.50}$$

is minimum. It is assumed that for every spanning tree T, $\sum_{(i,j)\,\varepsilon\, T} d_{ij} > 0$.

Chandrasekaran [14] has provided the following characterization of the MRST:

Theorem 2.4 ([14]): A spanning tree T^* is a MRST with minimum ratio as k^* if and only if T^* is a MST with the length of each arc $(i,j) \varepsilon A$ as $(c_{ij}-k^* d_{ij})$.

Using this characterization, the following method determines a MRST:

Let k^o be an arbitrary number. Determine the MST with the length of each arc $(i,j) \varepsilon A$ as $(c_{ij}-k^o d_{ij})$. Let T^o be the MST thus obtained. Let $M^o = \sum_{(i,j)\,\varepsilon\, T^o} (c_{ij}-k^o d_{ij})$. If $M^o = 0$, then T^o is a MRST. If $M^o < 0$, then treating the length of each arc $(i,j) \varepsilon A$ as $(c_{ij}-k\, d_{ij})$, perform the parametric analysis of the MST problem for

decreasing values of k until a spanning T^* is obtained for which $\sum_{(i,j) \in T^*} (c_{ij} - k\, d_{ij}) = 0$. The tree T^* is a MRST. If $M^o > 0$, then the parametric analysis is performed for increasing values of k.

It follows from Theorem 2.2 that this method is polynomially bounded.

2.6.2 Minimum Ratio Path Problem

The minimum ratio path problem is to determine a directed path P^* from source to sink for which the ratio $\sum_{(i,j) \in P^*} c_{ij} / \sum_{(i,j) \in P^*} d_{ij}$ is minimum. A characterization similar to Theorem 2.4, can be obtained for the MRP. The parametric analysis of the shortest path problem can be used to identify a MRP, but one serious difficulty arises. The parametric analysis assumes that the network does not contain any negative cycle for any value of the parameter in the range of the analysis. If the network is acyclic, this assumption is satisfied and the MRP is obtained by the parametric analysis. However, if the network is cyclic, negative cycles may arise during parametric analysis and the MRP may not be obtained. In fact, it can be shown that the MRP problem for cyclic networks is NP-complete.

Theorem 2.5: The minimum ratio path problem is NP-complete.

Proof : We show that the longest path problem, which is a wellknown NP-complete problem, is a special case of the MRP problem.

Associate a number ℓ_{ij} with each $(i,j) \in A$. The longest path is a directed path P^{**} from source to sink for which $\sum_{(i,j) \in P^{**}} \ell_{ij}$

is maximum. Define c_{ij} and d_{ij} for each $(i,j) \varepsilon A$ in the following manner:

$$c_{ij} = \begin{cases} 1 & , \text{ if } (i,j) \varepsilon O(s), \\ 0 & , \text{ otherwise}, \end{cases} \tag{2.51}$$

$$d_{ij} = \ell_{ij} \quad . \tag{2.52}$$

Clearly, for every directed path P from source to sink $\sum_{(i,j) \varepsilon P} c_{ij} = 1$. Thus, the MRP is a directed path P^* from source to sink for which the ratio $1/\sum_{(i,j) \varepsilon P^*} \ell_{ij}$ is minimum. The ratio $1/\sum_{(i,j) \varepsilon P^*} \ell_{ij}$ is minimum if and only if $\sum_{(i,j) \varepsilon P^*} \ell_{ij}$ is maximum. Hence $P^{**} = P^*$ and the theorem follows.

CHAPTER III

CONSTRAINED NETWORK CAPACITY EXPANSION PROBLEM

3.1 *INTRODUCTION*

The minimum cost flow problem is to find the optimum distribution of flow through the arcs of a given network from source to sink. The minimum cost flow problem subsumes a number of network problems, viz., the maximum flow problem, the shortest path problem, the assignment problem, the transportation problem and the transshipment problem.

In this chapter, we consider the minimum cost flow problem with an additional linear constraint. Such situations arise often in practice. Dantzig [22] points out that in the transportation problem, the additional constraint appears in the form of a lower or upper bounded partial sum of some of the variables. Charnes and Cooper [15] suggest that in many applications, the additional constraint may simply express the proportionality or equality of particular variables to force two routes to have equal flow. Other applications of the constrained minimum cost flow problem are the warehouse funds-flow model discussed by Charnes and Cooper [15] and a constrained version of Wagner's [92] employment scheduling problem.

The minimum cost flow problem is a specially structured linear program and this structure permits to achieve many computational simplifications while implementing simplex algorithm or its variants.

The additional linear constraint affects this structure and many of the nice features of the minimum cost flow problem are lost. For instance, the basis becomes nontriangular and the optimum solution may be nonintegral. However, the constraint matrix still possesses a structure and the attempt is to exploit this structure as far as possible.

If the additional constraint is of a particular form, attempts have been made to transform the problem into an equivalent minimum cost flow problem of larger size [15, 16, 17, 22, 60, 92]. However, these transformations are not possible with an arbitrary additional constraint. Klingman and Russell [62] developed a specialization of the primal simplex method for the constrained transportation problem, which was subsequently extended to the constrained transshipment problem by Glover et. al [43],and to the constrained generalized network problem by Hultz and Klingman [53] . All these methods consist of efficient ways to determine relative cost coefficients, representation of the entering arc and updating the basis. The efficiency of these methods have been established through extensive computational experience. The primal simplex approach is also extended to develop methods for multiconstrained (i) transportation problem [61]; capacitated transshipment problem [18] ; and (iii) generalized network problem [53] . Recently, Masch [65] has developed a cyclic method of solving the transshipment problem with an additional linear constraint. This method is not a simplex adaptation, but exploits the structure more efficiently.

Several researchers have considered the capacity expansion of a network where the budgetary constraint appears in the form of an additional linear constraint.

A common feature of physical distribution systems such as electricity or gas supply, road or rail networks is that the supply and demand requirements change with time. When the existing network can not accomodate the increased flow due to increasing demand requirements, expansion of the network is required. Sometimes, even if the supply and demand requirements do not change, expansion may be called for to reduce the transportation costs. In these situations, it is necessary to determine the optimum expansion of the network.

One way to expand the network is by increasing the capacities of various arcs. Several authors have considered the problem of optimally allocating a given budget to increase capacities of various arcs so that the flow from source to sink is maximized. Fulkerson [39] considers linear capacity expansion costs and proposes a labelling method to solve the problem. Hu [52] solves this problem by solving a sequence of shortest path problems. Bansal and Jacobson [9] consider concave expansion costs and approach the problem by Bender's decomposition procedure. When the cost functions are continuous and piecewise linear, the branch and bound algorithm discussed by Tomlin [89] can be applied. Price [79] has also developed a branch and bound algorithm to consider step functions.

An alternate way to expand the network is by introducing new arcs. Following this approach, Hammer [48] solves the problem of maximizing

flow from source to sink subject to the budgetary constraint by pseudo-boolean programming. Christofides and Brooker [19] describe an efficient tree search algorithm using bounds calculated by a dynamic programming procedure.

Some of the other problems, related to the optimum expansion of a network, are studied by Gomory and Hu [45] and Oettli and Prager [73].

In this chapter, we consider the constrained minimum cost flow problem and the network capacity expansion problem in a single model, which we term as the Constrained Network Capacity Expansion (CNCE) problem. The mathematical statement of the CNCE problem is as follows:

$$\text{Minimize} \quad Z = \sum_{(i,j)\in A} c_{ij}\, x_{ij}\,, \tag{3.1}$$

subject to

$$\sum_{(j,i)\in I(i)} x_{ji} - \sum_{(i,j)\in O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i\neq s,t, \quad \forall\, i\in N, \\ v, & \text{if } i=t, \end{cases} \tag{3.2}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij}, \ \forall\,(i,j)\in A, \tag{3.3}$$

$$0 \leq y_{ij} \leq a_{ij} - b_{ij}, \ \forall\,(i,j)\in A, \tag{3.4}$$

$$\sum_{(i,j)\in A} e_{ij}x_{ij} + \sum_{(i,j)\in A} f_{ij}y_{ij} \leq D, \tag{3.5}$$

where x_{ij} denotes the flow on arc (i,j) and y_{ij} denotes the amount by which the capacity of arc (i,j) is increased. We assume that $f_{ij} > 0$ for each $(i,j)\in A$.

As an application of the CNCE problem, consider a transportation network where c_{ij} and e_{ij} denote, respectively, the travel time and travel cost of transporting one unit of a commodity over arc (i,j). Each arc (i,j) has an existing capacity which may be increased by incurring some additional cost. Let $f_{ij} > 0$ denote the cost of increasing the capacity of arc (i,j) by one unit. The increase in the capacity is further limited by an amount $(a_{ij}-b_{ij})$. Then, the problem of minimizing the total time of transportation (or, equivalently, the average time of transportation) subject to the consideration that the cost of transportation and capacity expansion does not exceed a given budget D, is an instance of the CNCE problem.

We develop a parametric algorithm for the CNCE problem. Properties of the optimum solution of the CNCE problem and extensions of the concepts of bounded variable linear programs are used to evolve the concept of optimum basis structure for the CNCE problem. The optimum basis structure considers the variables y_{ij} and the constraints (3.3) and (3.4) implicitly, thereby reducing the size of the optimum basis considerably. Further, the constraint (3.5) is considered in a manner that preserves the basis triangularity.

The optimum basis structure is used to parametrize D. Initially, an optimum basis structure for a large value of D is obtained. The value of D is then subsequently decreased and an optimum basis structure is maintained at each step. It is obvious that a given basis structure is optimum for values of D contained in an interval.

The algorithm, in fact, determines such intervals and terminates when either the desired value of D is reached or infeasibility of the CNCE problem is discovered.

Although, the optimum solution of the CNCE problem may be non-integer, the algorithm performs almost all the computations in integers. Often practical considerations require integrality of the solution. To take care of such situations, we suggest a slight modification in the algorithm to obtain a near-optimum integer solution.

We show that the CNCE algorithm can be used to solve (i) a variation of the CNCE problem; and (ii) a bicriteria network problem. Several special cases of the CNCE problem are also considered and resulting simplifications in the CNCE algorithm are pointed out.

Lastly, we consider the capacity expansion of a capacitated transshipment network where supplies at source nodes and demands at sink nodes are linear functions of a single parameter λ. The problem considered is to determine the least cost of increasing arc capacities so that the changing demands are met by the changing supplies resulting from changes in λ. We develop an algorithm to solve this capacity expansion problem for all values of λ for which a feasible flow exists. A special case of this problem is also considered.

3.2 *OPTIMALITY CONDITIONS*

Let us modify the CNCE problem by replacing the constraint (3.5) by the following constraint:

$$- \sum_{(i,j) \in A} e_{ij} x_{ij} - \sum_{(i,j) \in A} f_{ij} y_{ij} = - D' . \tag{3.6}$$

We refer to the modified problem as the Modified Constrained Network Capacity Expansion (MCNCE) problem.

Let us partition A into the sets $\bar{X}$, $\bar{Y}$, L, U and V such that $\bar{X} \cup \bar{Y}$ consists of a spanning tree augmented by a nontree-arc. If we set

$$x_{ij} = 0 \quad , \quad \forall (i,j) \in L , \tag{3.7}$$

$$x_{ij} = b_{ij} \quad , \quad \forall (i,j) \in U , \tag{3.8}$$

$$x_{ij} = a_{ij} \quad , \quad \forall (i,j) \in V, \tag{3.9}$$

then there exists a unique solution, x_{ij} and y_{ij}, satisfying (3.2) and (3.6). If this solution further satisfies

$$0 \leq x_{ij} \leq b_{ij}, \quad \forall (i,j) \in \bar{X} , \tag{3.10}$$

$$b_{ij} \leq x_{ij} \leq a_{ij}, \quad \forall (i,j) \in \bar{Y} , \tag{3.11}$$

then we refer to $\bar{X} \cup \bar{Y}$ as a feasible basis and $(\bar{X} \cup \bar{Y}, L, U, V)$ as a feasible basis structure of the MCNCE problem. We refer to an arc (i,j) as an unsaturated basic arc if $(i,j) \in \bar{X}$ and a saturated basic arc if $(i,j) \in \bar{Y}$. We refer to the arcs belonging to L, U and V as the nonbasic arcs at their lower, middle and upper bounds respectively. We define an optimum basis structure of the MCNCE problem as a feasible basis structure for which the associated flow is an optimum solution of the MCNCE problem. In this section, we derive the necessary and sufficient conditions for a feasible basis structure to be an optimum basis structure.

The dual of the MCNCE problem is

$$\text{Maximize} \quad W = v(\pi_t - \pi_s) - \sum_{(i,j)\in A} b_{ij}\rho_{ij} - \sum_{(i,j)\in A} (a_{ij} - b_{ij})\,\delta_{ij} - \mu D\,, \tag{3.12}$$

subject to

$$\pi_j - \pi_i - \rho_{ij} - \mu e_{ij} \leq c_{ij}, \quad \forall\,(i,j) \in A, \tag{3.13}$$

$$\rho_{ij} - \mu f_{ij} - \delta_{ij} \leq 0 \quad , \quad \forall\,(i,j) \in A, \tag{3.14}$$

$$\rho_{ij} \geq 0 \quad \text{and} \quad \delta_{ij} \geq 0, \ \forall\,(i,j) \in A, \tag{3.15}$$

where π_i, ρ_{ij}, δ_{ij} and μ are the dual variables associated with the constraints (3.2), (3.3), (3.4) and (3.6) respectively.

The complementary slackness conditions for the MCNCE problem are

$$\rho_{ij}\,(b_{ij} + y_{ij} - x_{ij}) = 0, \quad \forall\,(i,j) \in A, \tag{3.16}$$

$$\delta_{ij}\,(a_{ij} - b_{ij} - y_{ij}) = 0, \quad \forall\,(i,j) \in A, \tag{3.17}$$

$$x_{ij}\,(c_{ij} + \mu e_{ij} + \rho_{ij} + \pi_i - \pi_j) = 0, \quad \forall\,(i,j) \in A, \tag{3.18}$$

$$y_{ij}\,(\delta_{ij} + \mu f_{ij} - \rho_{ij}) = 0, \quad \forall\,(i,j) \in A. \tag{3.19}$$

It is easy to see that the complementary slackness conditions for the CNCE problem are (3.16)-(3.19) and the following condition:

$$\mu(D - \sum_{(i,j)\in A} e_{ij}\,x_{ij} - \sum_{(i,j)\in A} f_{ij}\,y_{ij}) = 0. \tag{3.20}$$

This additional condition immediately yields the following relationship between the optimum solutions of the CNCE and MCNCE problem.

<u>Property 3.1</u> : The optimum solution of the MCNCE problem for $D' = D$ is the optimum solution of the CNCE problem.

We now derive an alternate representation of the complementary slackness conditions for the MCNCE problem. In the derivation, it is assumed that $\mu > 0$, which is maintained throughout the algorithm. We first prove the following result:

Theorem 3.1 : If $\mu > 0$, then any optimum solution, x_{ij} and y_{ij}, of the MCNCE problem satisfies

$$y_{ij}(b_{ij} + y_{ij} - x_{ij}) = 0 \, , \quad \forall (i,j) \in A. \tag{3.21}$$

Proof : Suppose $y_{ij} > 0$ and $x_{ij} < b_{ij} + y_{ij}$ for an arc (i,j). It follows from (3.16) and (3.19) that

$$x_{ij} < b_{ij} + y_{ij} \implies \rho_{ij} = 0, \tag{3.22}$$

$$y_{ij} > 0 \implies \rho_{ij} = \delta_{ij} + \mu f_{ij} \, . \tag{3.23}$$

Hence

$$x_{ij} < b_{ij} + y_{ij} \text{ and } y_{ij} > 0 \implies \delta_{ij} + \mu f_{ij} = 0. \tag{3.24}$$

Since $f_{ij} > 0$ and $\delta_{ij} \geq 0$, (3.24) implies that $\mu \leq 0$, which contradicts our assumption that $\mu > 0$.

Theorem 3.1 implies that in the optimum solution of the MCNCE problem if $y_{ij} > 0$, then $x_{ij} = b_{ij} + y_{ij}$; and if $x_{ij} < b_{ij} + y_{ij}$, then $y_{ij} = 0$. Using these properties, the conditions (3.16)-(3.19) can be shown equivalent to

$$0 < x_{ij} < b_{ij} \implies \pi_j - \pi_i = c_{ij} + \mu\, e_{ij}, y_{ij} = 0, \ \forall (i,j) \in A, \tag{3.25}$$

$$b_{ij} < x_{ij} < a_{ij} \implies \pi_j - \pi_i = c_{ij} + \mu(e_{ij} + f_{ij}), y_{ij} = x_{ij} - b_{ij}, \quad \forall (i,j) \in A, \tag{3.26}$$

$$\pi_j - \pi_i < c_{ij} + \mu e_{ij} \Rightarrow x_{ij} = 0,\ y_{ij} = 0,\ \forall\,(i,j) \in A, \tag{3.27}$$

$$c_{ij} + \mu e_{ij} < \pi_j - \pi_i < c_{ij} + \mu(e_{ij} + f_{ij}) \Rightarrow x_{ij} = b_{ij}, y_{ij} = 0,\ \forall\,(i,j) \in A, \tag{3.28}$$

$$\pi_j - \pi_i > c_{ij} + \mu(e_{ij} + f_{ij}) \Rightarrow x_{ij} = a_{ij},\ y_{ij} = a_{ij} - b_{ij},\ \forall\,(i,j) \in A. \tag{3.29}$$

It is clear from (3.25)-(3.29) that every optimum solution satisfies

$$y_{ij} = \max.\{0, x_{ij} - b_{ij}\},\ \forall\,(i,j) \in A. \tag{3.30}$$

This relationship indicates that in the optimum solution, y_{ij} can be considered implicitly. In view of this relationship, we restrict our attention to x_{ij} only. It further follows from (3.25)-(3.29) that the necessary and sufficient conditions for a feasible basis structure $(\bar{X} \cup \bar{Y}, L, U, V)$ to be an optimum basis structure are that there exist numbers π_j satisfying the following conditions:

$$\pi_j - \pi_i = c_{ij} + \mu e_{ij},\ \forall\,(i,j) \in \bar{X}, \tag{3.31}$$

$$\pi_j - \pi_i = c_{ij} + \mu(e_{ij} + f_{ij}),\ \forall\,(i,j) \in \bar{Y}, \tag{3.32}$$

$$\pi_j - \pi_i \le c_{ij} + \mu e_{ij},\ \forall\,(i,j) \in L, \tag{3.33}$$

$$c_{ij} + \mu e_{ij} \le \pi_j - \pi_i \le c_{ij} + \mu(e_{ij} + f_{ij}),\ \forall\,(i,j) \in U, \tag{3.34}$$

$$\pi_j - \pi_i \ge c_{ij} + \mu(e_{ij} + f_{ij}),\ \forall\,(i,j) \in V. \tag{3.35}$$

The set of basic arcs, $\bar{X} \cup \bar{Y}$, consists of a spanning tree, say T, and an additional arc, say (p,q). Let $X = \bar{X} \cap T$ and $Y = \bar{Y} \cap T$.

Consequently, we may represent $\bar{X} \cup \bar{Y}$ as $X \cup Y \cup \{(p,q)\}$. We refer to $X \cup Y$ as the basic tree and arcs belonging to it as tree-arcs. Arcs not belonging to the basic tree are referred to as nontree-arcs. Further, we refer to (p,q) as the pivot arc. It may be remembered that the pivot arc can be an unsaturated or saturated basic arc. We now use the special structure of the basis to simplify (3.31)-(3.35).

Let π_j^c, π_j^d and π_j be the numbers satisfying

$$\pi_s^c = 0 \quad \text{and} \quad \pi_j^c - \pi_i^c = c_{ij}, \ \forall (i,j) \in X \cup Y, \tag{3.36}$$

$$\pi_s^d = 0 \quad \text{and} \quad \pi_j^d - \pi_i^d = \begin{cases} e_{ij} & , \ \forall (i,j) \in X, \\ e_{ij} + f_{ij}, & \forall (i,j) \in Y, \end{cases} \tag{3.37}$$

$$\pi_s = 0 \quad \text{and} \quad \pi_j - \pi_i = \begin{cases} c_{ij} + \mu e_{ij}, & \forall (i,j) \in X, \\ c_{ij} + \mu(e_{ij} + f_{ij}), & \forall (i,j) \in Y, \end{cases} \tag{3.38}$$

where μ is a positive real number.

It is clearly seen that

$$\pi_j = \pi_j^c + \mu \pi_j^d \ , \quad \forall j \in N. \tag{3.39}$$

Let $x_{pq} = 0$ or b_{pq} or a_{pq}. Let $\bar{L}$, $\bar{U}$ and $\bar{V}$ be the sets of nontree arcs having $x_{ij} = 0$, b_{ij} and a_{ij} respectively. For each arc $(k,l) \in \bar{L} \cup \bar{U}$, define the numbers $\bar{c}_{kl}$ and $\bar{d}_{kl}$ as follows:

$$\bar{c}_{kl} = \begin{cases} \pi_k^c - \pi_l^c + c_{kl} & , \text{ if } (k,l) \in \bar{L} , \\ \pi_l^c - \pi_k^c - c_{kl} & , \text{ if } (k,l) \in \bar{U} , \end{cases} \tag{3.40}$$

$$\bar{d}_{kl} = \begin{cases} \pi_k^d - \pi_l^d + e_{kl} \quad , & \text{if} \quad (k,l) \in \bar{L} , \\ \pi_l^d - \pi_k^d - e_{kl} \quad , & \text{if} \quad (k,l) \in \bar{U} \end{cases} \tag{3.41}$$

Similarly, for each arc $(k,l) \in \bar{U} \cup \bar{V}$, define the numbers $\underline{c}_{kl}$ and $\underline{d}_{kl}$ as follows:

$$\underline{c}_{kl} = \begin{cases} \pi_k^c - \pi_l^c + c_{kl} , & \text{if} \quad (k,l) \in \bar{U}, \\ \pi_l^c - \pi_k^c - c_{kl} , & \text{if} \quad (k,l) \in \bar{V}, \end{cases} \tag{3.42}$$

$$\underline{d}_{kl} = \begin{cases} \pi_k^d - \pi_l^d + e_{kl} + f_{kl}, & \text{if} \quad (k,l) \in \bar{U}, \\ \pi_l^d - \pi_k^d - e_{kl} - f_{kl}, & \text{if} \quad (k,l) \in \bar{V}. \end{cases} \tag{3.43}$$

It immediately follows from (3.40)-(3.43) that

$$\underline{c}_{kl} = -\bar{c}_{kl} \text{ and } \underline{d}_{kl} = -\bar{d}_{kl} + f_{kl}, \ \forall \ (k,l) \in \bar{U} \tag{3.44}$$

Substituting (3.39) in (3.31)-(3.35), and then using (3.40)-(3.43), we obtain the following equivalent conditions:

$$\begin{cases} \bar{c}_{pq} + \mu \bar{d}_{pq} = 0, \text{ if } (p,q) \text{ is an unsaturated basic arc,} \\ \underline{c}_{pq} + \mu \underline{d}_{pq} = 0, \text{ if } (p,q) \text{ is a saturated basic arc,} \end{cases} \tag{3.45}$$

$$\bar{c}_{kl} + \mu \, \bar{d}_{kl} \geq 0, \ \forall \ (k,l) \in L \cup U, \tag{3.46}$$

$$\underline{c}_{kl} + \mu \, \underline{d}_{kl} \geq 0, \ \forall \ (k,l) \in U \cup V. \tag{3.47}$$

We refer to the conditions (3.45)-(3.47) as the <u>optimality conditions</u> for the MCNCE problem. It is clear from (3.45) that given the basic tree, the value of μ depends upon the arc (p,q). We refer μ as the <u>pivot ratio</u>.

3.3 DEVELOPMENT OF THE ALGORITHM

In this section, we describe in detail the development of the algorithm for the CNCE problem. The algorithm treats D' as a parameter and solves the MCNCE problem for continuously decreasing values of D'. It first obtains an optimum basis structure of the MCNCE problem for a sufficiently large value of D', say D_o. It then determines D_1 such that this basis structure continues to remain optimum for all $D' \varepsilon (D_1, D_o)$. A dual simplex iteration is performed for $D' = D_1$ to obtain an alternate optimum basis structure which may allow further decrease in D'. The algorithm proceeds in this manner until either an optimum solution of the CNCE problem is obtained or its infeasibility is indicated.

Given an optimum basis structure $(X \cup Y \cup \{(p,q)\}, L, U, V)$, let us classify the nontree-arcs, $\bar{L} \cup \bar{U} \cup \bar{V}$, as follows:

$$\bar{S} = \{(k,l) \varepsilon \bar{L} \cup \bar{U} : \bar{d}_{kl} < 0 \text{ and } \bar{c}_{kl} > 0\} , \tag{3.48}$$

$$\underline{S} = \{(k,l) \varepsilon \bar{U} \cup \bar{V} : \underline{d}_{kl} < 0 \text{ and } \underline{c}_{kl} > 0\} , \tag{3.49}$$

$$\bar{G} = \{(k,l) \varepsilon \bar{L} \cup \bar{U} : \bar{d}_{kl} < 0 \text{ and } \bar{c}_{kl} \leq 0\} , \tag{3.50}$$

$$\underline{G} = \{(k,l) \varepsilon \bar{U} \cup \bar{V} : \underline{d}_{kl} < 0 \text{ and } \underline{c}_{kl} \leq 0\} , \tag{3.51}$$

$$H = \{(k,l) \varepsilon \bar{L} \cup \bar{U} : \bar{d}_{kl} \geq 0\} \cup \{(k,l) \varepsilon \bar{U} \cup \bar{V} : \underline{d}_{kl} \geq 0\} \tag{3.52}$$

We refer to the sets $\bar{S} \cup \underline{S}$, $\bar{G} \cup \underline{G}$ and H as the sets of <u>active</u>, <u>critical</u> and <u>passive arcs</u> respectively. These sets are mutually exclusive as well as exhaustive.

3.3.1 Characteristic Interval

Let the optimum basis structure of the MCNCE problem for $D' = \bar{D}$ be $(X \cup Y \cup \{(p,q)\}, L, U, V)$. Here $\bar{D}$ is such that $(X \cup Y \cup \{(p,q)\}, L, U, V)$ is not an optimum basis structure for $D' > \bar{D}$. Let $\bar{x}_{ij}$ be the optimum flow and $\bar{Z} = \sum_{(i,j) \in A} c_{ij} \bar{x}_{ij}$. It will be shown in Sections 3.3.2 and 3.3.3 that if (p,q) is an unsaturated basic arc, then $(p,q) \in \bar{S}$, and $x_{pq} = 0$ or b_{pq}. It will also be shown there that if (p,q) is a saturated basic arc, then $(p,q) \in \underline{S}$, and $x_{pq} = b_{pq}$ or a_{pq}. We now determine the interval $(\underline{D}, \bar{D})$ such that the current basis structure continues to remain optimum for all $D \in (\underline{D}, \bar{D})$. This interval is known as the <u>characteristic interval</u> associated with $(X \cup Y \cup \{(p,q)\}, L, U, V)$.

Since the numbers $\bar{c}_{kl}$, $\bar{d}_{kl}$, $\underline{c}_{kl}$ and $\underline{d}_{kl}$ are uniquely determined for a given basis structure, the optimality conditions are not affected by change in the value of D'. However, in order to satisfy (3.6), flow must be changed keeping the basis structure intact.

In the basis, there is a cycle consisting of basic arcs which is formed when (p,q) is added to the basic tree. Let W_{pq} be the set of arcs in this cycle. The flow can be changed by circulating some additional flow in W_{pq}. The direction of additional flow in W_{pq} depends upon the status of arc (p,q) which could be either saturated or unsaturated. If (p,q) is an unsaturated basic arc, then $(p,q) \in \bar{S} \subseteq (\bar{L} \cup \bar{U})$. Hence to keep the arc (p,q) unsaturated, the additional flow must be circulated along (p,q) if $(p,q) \in \bar{L}$ and opposite

to (p,q) if $(p,q) \in \bar{U}$. Similarly, if (p,q) is a saturated basic arc, then $(p,q) \in \underline{S} \subseteq (\bar{U} \cup \bar{V})$. This implies that the additional flow must be circulated along (p,q) if $(p,q) \in \bar{U}$ and opposite to (p,q) if $(p,q) \in \bar{V}$. After defining the orientation of W_{pq} accordingly, let $\bar{W}_{pq}$ and $\underline{W}_{pq}$ be the sets of arcs in W_{pq} along and opposite to its orientation respectively. The maximum amount of flow, denoted by $\bar{w}$, that can be sent in W_{pq} is calculated by using the bound restrictions of the basic arcs, i.e., (3.10) and (3.11). Let us define the number $\bar{w}_{ij}$ for each $(i,j) \in W_{pq}$ as follows:

$$\bar{w}_{ij} = \begin{cases} b_{ij} - \bar{x}_{ij}, & \text{if } (i,j) \in \bar{X} \cap \bar{W}_{pq}, \\ \bar{x}_{ij}, & \text{if } (i,j) \in \bar{X} \cap \underline{W}_{pq}, \\ a_{ij} - \bar{x}_{ij}, & \text{if } (i,j) \in \bar{Y} \cap \bar{W}_{pq}, \\ \bar{x}_{ij} - b_{ij}, & \text{if } (i,j) \in \bar{Y} \cap \underline{W}_{pq}, \end{cases} \tag{3.53}$$

where $\bar{X}$ and $\bar{Y}$ are the sets of unsaturated and saturated basic arcs in $X \cup Y \cup \{(p,q)\}$ respectively.

The $\bar{w}_{ij}$ denotes the maximum additional flow permitted by any arc $(i,j) \in W_{pq}$ without violating its bound restrictions. Hence

$$\bar{w} = \min_{(i,j) \in W_{pq}} \{ \bar{w}_{ij} \}. \tag{3.54}$$

Let

$$\underline{D} = \begin{cases} \bar{D} + \bar{w}\, \bar{d}_{pq}, & \text{if } (p,q) \text{ is an unsaturated basic arc,} \\ \bar{D} + \bar{w}\, \underline{d}_{pq}, & \text{if } (p,q) \text{ is a saturated basic arc.} \end{cases} \tag{3.55}$$

The following theorem establishes the optimum flow of the MCNCE problem for all $D' \in (\underline{D}, \bar{D})$.

<u>Theorem 3.2</u> : For all $D' \in (\underline{D}, \bar{D})$, the optimum flow of the MCNCE problem is given by

$$x_{ij} = \begin{cases} \bar{x}_{ij} + \theta \, \bar{w} \,, & \forall \, (i,j) \in \bar{W}_{pq} \,, \\ \bar{x}_{ij} - \theta \, \bar{w} \,, & \forall \, (i,j) \in \underline{W}_{pq} \,, \\ \bar{x}_{ij} \,, & \forall \, (i,j) \notin W_{pq} \,, \end{cases} \tag{3.56}$$

and the value of the objective function is

$$Z = \begin{cases} \bar{Z} + \theta \, \bar{c}_{pq} \, \bar{w} \,, & \text{if } (p,q) \text{ is an unsaturated basic arc,} \\ \bar{Z} + \theta \, \underline{c}_{pq} \, \bar{w} \,, & \text{if } (p,q) \text{ is a saturated basic arc.} \end{cases} \tag{3.57}$$

where

$$\theta = (\bar{D} - D')/(\bar{D} - \underline{D}). \tag{3.58}$$

<u>Proof</u> : Let (p,q) be an unsaturated basic arc. It can be easily shown using (3.36) and (3.40) that

$$\bar{c}_{pq} = \sum_{(i,j) \in \bar{W}_{pq}} c_{ij} - \sum_{(i,j) \in \underline{W}_{pq}} c_{ij} \,. \tag{3.59}$$

Similarly, using (3.37) and (3.41) it can be shown that

$$\bar{d}_{pq} = \Big(\sum_{(i,j) \in \bar{W}_{pq}} e_{ij} - \sum_{(i,j) \in \underline{W}_{pq}} e_{ij} \Big) + \Big(\sum_{(i,j) \in \bar{W}_{pq} \cap \bar{Y}} f_{ij} - \sum_{(i,j) \in \underline{W}_{pq} \cap \bar{Y}} f_{ij} \Big). \tag{3.60}$$

Hence $\bar{c}_{pq}$ and $\bar{d}_{pq}$ denote, respectively, the increase in the value of Z and D' if unit amount of additional flow is

circulated in W_{pq} along its orientation. Since $(p,q) \varepsilon \bar{S}$, we have $\bar{d}_{pq} < 0$ and $\bar{c}_{pq} > 0$. Therefore Z increases and D' decreases as additional flow is sent in W_{pq}. Since $\mu = -\bar{c}_{pq}/\bar{d}_{pq} > 0$, it denotes the increase in Z per unit decrease in D'.

For $D' \varepsilon (\underline{D},\bar{D})$, we see that $0 \leq \theta \leq 1$ and hence the flow x_{ij}, given by (3.56), satisfies the flow requirements of the optimum basis structure. We now show that the condition (3.6) remains satisfied in the interval $(\underline{D},\bar{D})$. Substituting x_{ij} from (3.56) in (3.6),

$$\text{R.H.S. of (3.6)} = -\sum_{(i,j)\,\varepsilon\, A} e_{ij}\,\bar{x}_{ij} - \theta\,\bar{w}\Big(\sum_{(i,j)\,\varepsilon\,\bar{W}_{pq}} e_{ij} - \sum_{(i,j)\,\varepsilon\,\underline{W}_{pq}} e_{ij}\Big) - \theta\,\bar{w}\Big(\sum_{(i,j)\,\varepsilon\,\bar{W}_{pq}\cap\bar{Y}} f_{ij} - \sum_{(i,j)\,\varepsilon\,\underline{W}_{pq}\cap\bar{Y}} f_{ij}\Big) \tag{3.61}$$

$$= -\bar{D} - \theta\,\bar{w}\,\bar{d}_{pq} \quad \text{(using (3.60))} \tag{3.62}$$

$$= -D' \qquad \text{(using (3.55) and (3.58))} \tag{3.63}$$

This proves the optimality of x_{ij}. To prove (3.57) consider

$$\text{R.H.S. of (3.57)} = \sum_{(i,j)\,\varepsilon\, A} c_{ij}\,\bar{x}_{ij} + \theta\,\bar{w}\Big(\sum_{(i,j)\,\varepsilon\,\bar{W}_{pq}} c_{ij} - \sum_{(i,j)\,\varepsilon\,\underline{W}_{pq}} c_{ij}\Big) \tag{3.64}$$

$$= \bar{Z} + \theta\,\bar{w}\,\bar{c}_{pq} \quad \text{(using (3.59))} \tag{3.65}$$

The proof of the theorem is given for the case when (p,q) is an unsaturated basic arc. If (p,q) is a saturated basic arc, then $\underline{c}_{pq}$ and $\underline{d}_{pq}$ denote, respectively, the increase in the value of Z and D' when unit amount of additional flow is circulated in W_{pq}

along its orientation. The interpretation of μ remains the same. Further, optimality of x_{ij} and Z can be proved in the similar manner.

Let $\underline{x}_{ij}$ be the optimum flow for $D' = \underline{D}$. Further decrease in the value of D' is blocked by an arc (α, β) for which $\bar{w}_{\alpha\beta} = \bar{w}$. If $(\alpha, \beta) \varepsilon \bar{X}$, then $\underline{x}_{\alpha\beta} = 0$ or $b_{\alpha\beta}$; and if $(\alpha, \beta) \varepsilon \bar{Y}$, then $\underline{x}_{\alpha\beta} = b_{\alpha\beta}$ or $a_{\alpha\beta}$. If D' is decreased without changing the optimum basis structure, flow in arc (α, β) violates its bounds. Thus a dual simplex iteration is performed to obtain an alternate optimum basis structure for $D' = \underline{D}$, which may allow further decrease in the value of D'.

3.3.2 Dual Simplex Iteration

The alternate optimum basis structure for $D' = \bar{D}$ is obtained by performing a dual simplex iteration. All the parameters and sets associated with the optimum basis structure $(X \cup Y \cup \{(p,q)\}, L, U, V)$ are indicated with a prime for the alternate optimum basis structure.

The dual simplex iteration is performed by dropping the arc (α, β) from the basis and selecting a nonbasic arc to enter the basis. When the arc (α, β) is dropped from the basis, the remaining basic arcs constitute a spanning tree. This spanning tree becomes the basic tree in the alternate optimum basis structure. If $(\alpha, \beta) = (p,q)$, then $X' = X$ and $Y' = Y$. In this case, the values of $\bar{c}'_{kl}$, $\bar{d}'_{kl}$, $\underline{c}'_{kl}$ and $\underline{d}'_{kl}$ remain unchanged for all nontree arcs other than the arc (p,q). The values of $\bar{c}_{pq}$, $\bar{d}_{pq}$, $\underline{c}_{pq}$ and $\underline{d}_{pq}$ are multiplied by -1 to obtain

$\bar{c}'_{pq}$, $\bar{d}'_{pq}$, $\underline{c}'_{pq}$ and $\underline{d}'_{pq}$. If $(\alpha,\beta) \neq (p,q)$, then the basic tree changes. The new basic tree is given by $X' \cup Y' = X \cup Y \cup \{(p,q)\} - \{(\alpha,\beta)\}$. This basic tree consists of two subtrees T'_1 (containing source) and T'_2, and the arc (p,q) which joins these two subtrees. The values of $\bar{c}'_{kl}$, $\bar{d}'_{kl}$, $\underline{c}'_{kl}$ and $\underline{d}'_{kl}$ also get changed. To compute these values, let us classify the nontree arcs $\bar{L}' \cup \bar{U}' \cup \bar{V}'$ as follows:

$$E^0 = \{(k,l) \in \bar{L}' \cup \bar{U}' \cup \bar{V}' : k \in T'_1 \text{ and } l \in T'_1; \text{ or } k \in T'_2 \text{ and } l \in T'_2\}, \tag{3.66}$$

$$E^1 = \{(k,l) \in \bar{L}' \cup \bar{U}' \cup \bar{V}' : k \in T'_1, l \in T'_2 \text{ and } (k,l) \in \bar{L}' \cup \bar{V}'; \text{ or } k \in T'_2, l \in T'_1 \text{ and } (k,l) \in \bar{U}'\} \tag{3.67}$$

$$E^2 = \{(k,l) \in \bar{L}' \cup \bar{U}' \cup \bar{V}' : k \in T'_1, l \in T'_2 \text{ and } (k,l) \in \bar{U}'; \text{ or } k \in T'_2, l \in T'_1 \text{ and } (k,l) \in \bar{L}' \cup \bar{V}'\}. \tag{3.68}$$

Using (3.36)-(3.37) and (3.40)-(3.43) it can be easily shown that $\bar{c}'_{kl}$, $\bar{d}'_{kl}$, $\underline{c}'_{kl}$ and $\underline{d}'_{kl}$ can be expressed in terms of $\bar{c}_{kl}$, $\bar{d}_{kl}$, $\underline{c}_{kl}$ and $\underline{d}_{kl}$, depending upon the status of the arc (p,q). These expressions are as follows:

<u>Case I</u> : (p,q) is an unsaturated basic arc

$$\bar{c}'_{kl} = \begin{cases} \bar{c}_{kl}, & \forall (k,l) \in E^0, \\ \bar{c}_{kl} - \bar{c}_{pq}, & \forall (k,l) \in E^1, \\ \bar{c}_{kl} + \bar{c}_{pq}, & \forall (k,l) \in E^2, \end{cases} \tag{3.69}$$

$$\bar{d}'_{kl} = \begin{cases} \bar{d}_{kl}, & \forall (k,l) \in E^0, \\ \bar{d}_{kl}-\bar{d}_{pq}, & \forall (k,l) \in E^1, \\ \bar{d}_{kl}+\bar{d}_{pq}, & \forall (k,l) \in E^2, \end{cases} \tag{3.70}$$

$$\underline{c}'_{kl} = \begin{cases} \underline{c}_{kl}, & \forall (k,l) \in E^0, \\ \underline{c}_{kl}+\bar{c}_{pq}, & \forall (k,l) \in E^1, \\ \underline{c}_{kl}-\bar{c}_{pq}, & \forall (k,l) \in E^2, \end{cases} \tag{3.71}$$

$$\underline{d}'_{kl} = \begin{cases} \underline{d}_{kl}, & \forall (k,l) \in E^0, \\ \underline{d}_{kl}+\bar{d}_{pq}, & \forall (k,l) \in E^1, \\ \underline{d}_{kl}-\bar{d}_{pq}, & \forall (k,l) \in E^2. \end{cases} \tag{3.72}$$

Case II: (p,q) is a saturated basic arc. In (3.69)-(3.72), replacing $\bar{c}_{pq}$ and $\bar{d}_{pq}$ by $\underline{c}_{pq}$ and $\underline{d}_{pq}$, respectively, we get the desired expressions for $\bar{c}'_{kl}$, $\bar{d}'_{kl}$, $\underline{c}'_{kl}$ and $\underline{d}'_{kl}$.

Let $\bar{S}' \cup \underline{S}'$ denote the set of active arcs. If $\bar{S}' \cup \underline{S}'$ is empty, it follows from Theorem 3.6 that the MCNCE problem is infeasible for all $D' < \underline{D}$. However, if $\bar{S}' \cup \underline{S}'$ is nonempty, define a number μ'_{kl} for each $(k,l) \in \bar{S}' \cup \underline{S}'$ as follows:

$$\mu'_{kl} = \begin{cases} -\bar{c}'_{kl}/\bar{d}'_{kl}, & \text{if } (k,l) \in \bar{S}', \\ -\underline{c}'_{kl}/\underline{d}'_{kl}, & \text{if } (k,l) \in \underline{S}'. \end{cases} \tag{3.73}$$

Clearly, $\mu'_{kl} > 0$ for each $(k,l) \in \bar{S}' \cup \underline{S}'$. Let $(g,h) \in \bar{S}' \cup \underline{S}'$ be an arc satisfying

$$\mu'_{gh} = \min_{(k,l) \in \bar{S}' \cup \underline{S}'} \{\mu'_{kl}\}. \tag{3.74}$$

Further, let L', U' and V' be the sets of arcs in $\bar{L}' \cup \bar{U}' \cup \bar{V}' - \{(g,h)\}$ at their lower, middle and upper bounds respectively. We now state some theorems along with their proofs to show that $(X' \cup Y' \cup \{(g,h)\}, L', U', V')$ is an alternate optimum basis structure for $D' = \underline{D}$ with (g,h) as an unsaturated basic arc if $(g,h) \in \bar{S}'$, and as a saturated basic arc if $(g,h) \in \underline{S}'$. In Theorems 3.3, 3.4 and 3.5, it is assumed that (p,q) is an unsaturated basic arc. On similar lines, results can be proved with (p,q) as a saturated basic arc.

Theorem 3.3. No arc $(k,l) \in \bar{L}' \cup \bar{U}' \cup \bar{V}'$ is critical.

Proof. If $(k,l) \in E^0$, then substituting (3.69) and (3.70) in (3.46) we get

$$\bar{c}'_{kl} + \mu \bar{d}'_{kl} \geq 0. \tag{3.75}$$

If $(k,l) \in E^1$, then substracting (3.45) from (3.46) and using (3.69) and (3.70) we get (3.75). If $(k,l) \in E^2$, then adding (3.45) to (3.46) and using (3.69) and (3.70) we again get (3.75). We have thus obtained

$$\bar{c}'_{kl} + \mu \bar{d}'_{kl} \geq 0, \quad \forall (k,l) \in \bar{L}' \cup \bar{U}' . \tag{3.76}$$

Similarly, it can be shown that

$$\underline{c}'_{kl} + \mu \underline{d}'_{kl} \geq 0, \quad \forall (k,l) \in \bar{U}' \cup \bar{V}'. \tag{3.77}$$

If an arc $(k,l) \in \bar{G}'$ and is critical, then $(k,l) \in (\bar{L}' \cup \bar{U}')$ and $\bar{d}'_{kl} < 0$, $\bar{c}'_{kl} \leq 0$. Since $\mu > 0$, we have $\bar{c}'_{kl} + \mu \bar{d}'_{kl} < 0$ which contradicts (3.76). Likewise, if an arc $(k,l) \in \underline{G}'$ and is critical, then $(k,l) \in (\bar{U}' \cup \bar{V}')$ and $\underline{d}'_{kl} < 0$, $\underline{c}'_{kl} \leq 0$. Thus,

$\underline{c}'_{kl} + \mu\, \underline{d}'_{kl} < 0$ which contradicts (3.77).

Let μ' denote the value of the pivot ratio for the alternate optimum basis structure. It follows from (3.45) that

$$\mu' = \begin{cases} -\bar{c}'_{gh}/\bar{d}'_{gh}, & \text{if } (g,h) \text{ is an unsaturated basic arc,} \\ -\underline{c}'_{gh}/\underline{d}'_{gh}, & \text{if } (g,h) \text{ is a saturated basic arc.} \end{cases} \tag{3.78}$$

Comparing (3.78) with (3.73) we find that

$$\mu' = \mu'_{gh} \;. \tag{3.79}$$

Thus the pivot arc is an active arc which yields the lowest value of the pivot ratio. We now show that the value of pivot ratio is increased by the dual simplex iteration.

<u>Theorem 3.4</u> : $$\mu' \geq \mu \;. \tag{3.80}$$

<u>Proof</u> : Consider any arc $(k,l) \in \bar{S}' \cup \underline{S}'$. If $(k,l) \in \bar{S}'$, then (3.76) can be written as

$$\mu \leq -\bar{c}'_{kl}/\,\bar{d}'_{kl} = \mu'_{kl} \;. \tag{3.81}$$

If $(k,l) \in \underline{S}'$, then (3.77) can be written as

$$\mu \leq -\,\underline{c}'_{kl}/\underline{d}'_{kl} = \mu'_{kl}. \tag{3.82}$$

We have thus obtained

$$\mu \leq \mu'_{kl} \,,\ \forall\,(k,l) \in \bar{S}' \cup \underline{S}'. \tag{3.83}$$

Since $(g,h) \in \bar{S}' \cup \underline{S}'$, we have

$$\mu \leq \mu'_{gh} = \mu' \;. \tag{3.84}$$

Theorem 3.5 : The basis structure $(X' \cup Y' \cup \{(g,h)\}, L', U', V')$ satisfies the optimality conditions.

Proof : The dual simplex iteration is essentially a degerate iteration and flow on any arc does not change. Therefore the value of D' is not affected by this iteration. The optimality condition (3.45) is satisfied in view of (3.78). To prove that the remaining conditions are also satisfied, consider any arc $(k,l) \in \bar{S}' \cup \underline{S}'$. Clearly

$$\mu' \leq \mu'_{kl} = \begin{cases} -\bar{c}'_{kl}/\bar{d}'_{kl}, & \text{if } (k,l) \in \bar{S}', \\ -\underline{c}'_{kl}/\underline{d}'_{kl}, & \text{if } (k,l) \in \underline{S}', \end{cases} \tag{3.85}$$

which after rearrangement reduces to (3.46) and (3.47). Now consider any arc $(k,l) \notin \bar{S}' \cup \underline{S}'$. Since there is no critical arc, (k,l) must be a passive arc. Hence $\bar{d}'_{kl} \geq 0$ if $(k,l) \in \bar{L}' \cup \bar{U}'$, and $\underline{d}'_{kl} \geq 0$ if $(k,l) \in \bar{U}' \cup \bar{V}'$. Multiplying $\mu' \geq \mu$ by $\bar{d}'_{kl}$ and adding $\bar{c}'_{kl}$ to both the sides, we get

$$\bar{c}'_{kl} + \mu' \bar{d}'_{kl} \geq \bar{c}'_{kl} + \mu \bar{d}'_{kl}, \tag{3.86}$$

which reduces to (3.46) using (3.76).

Similarly, multiplying $\mu' \geq \mu$ by $\underline{d}'_{kl}$ and adding $\underline{c}'_{kl}$ to both the sides, we get

$$\underline{c}'_{kl} + \mu' \underline{d}'_{kl} \geq \underline{c}'_{kl} + \mu \underline{d}'_{kl}, \tag{3.87}$$

which reduces to (3.47) using (3.77). This completes the proof.

Theorem 3.6: If $\bar{S}' \cup \underline{S}'$ is empty, the MCNCB problem is infeasible for all $D' < \underline{D}$.

Proof : Let D_{min} be the minimum value of D' for which the MCNCE problem is feasible. Clearly, D_{min} is the optimum objective function value of the following linear program:

$$\text{Minimize} \quad \sum_{(i,j)\in A} e_{ij}x_{ij} + \sum_{(i,j)\in A} f_{ij}\, y_{ij}\ , \tag{3.88}$$

subject to

$$\sum_{(j,i)\in I(i)} x_{ji} - \sum_{(i,j)\in O(i)} x_{ij} = \begin{cases} -v\ , \text{ if } i=s, \\ 0\ , \text{ if } i\neq s,t,\ \forall\ i\in N, \\ v\ , \text{ if } i=t, \end{cases} \tag{3.89}$$

$$0 \le x_{ij} \le b_{ij} + y_{ij}\ ,\ \forall\ (i,j)\in A, \tag{3.90}$$

$$0 \le y_{ij} \le a_{ij} - b_{ij}\ ,\ \forall\ (i,j)\in A. \tag{3.91}$$

The dual of (3.88)-(3.91) is

$$\text{Maximize}\ (\pi_t - \pi_s)v - \sum_{(i,j)\in A} b_{ij}\rho_{ij} - \sum_{(i,j)\in A} (a_{ij}-b_{ij})\delta_{ij}, \tag{3.92}$$

subject to

$$\pi_j - \pi_i - \rho_{ij} \le e_{ij},\ \forall\ (i,j)\in A\ , \tag{3.93}$$

$$\rho_{ij} - \delta_{ij} \le f_{ij}\ ,\ \forall\ (i,j)\in A\ , \tag{3.94}$$

$$\rho_{ij} \ge 0 \text{ and } \delta_{ij} \ge 0,\ \forall\ (i,j)\in A, \tag{3.95}$$

and the complementary slackness conditions are

$$\rho_{ij}(b_{ij}+y_{ij}-x_{ij}) = 0,\ \forall\ (i,j)\in A, \tag{3.96}$$

$$\delta_{ij}(a_{ij}-b_{ij}-y_{ij}) = 0,\ \forall\ (i,j)\in A, \tag{3.97}$$

$$x_{ij}(e_{ij}+\rho_{ij}+\pi_i-\pi_j)=0,\ \forall\ (i,j)\in A, \tag{3.98}$$

$$y_{ij}(f_{ij}+\delta_{ij}-\rho_{ij}) = 0,\ \forall\ (i,j)\in A. \tag{3.99}$$

We now show that if $\bar{S}' \cup \underline{S}'$ is empty, the solution $\underline{x}_{ij}$ and $\underline{y}_{ij} = \max. \{0, \underline{x}_{ij} - b_{ij}\}$ solves (3.88)-(3.91) and hence $\underline{D} = D_{min}$. Let π_j^d be defined by (3.37). Set $\pi_j = \pi_j^d$, $\forall\, j \in N$. Hence

$$\pi_i - \pi_j + e_{ij} = 0 \quad , \forall\,(i,j) \in X', \tag{3.100}$$

$$\pi_i - \pi_j + e_{ij} = -f_{ij} \;, \forall\,(i,j) \in Y'. \tag{3.101}$$

Since the set of active arcs is empty and there is no critical arc, all the nontree-arcs are passive arcs. Hence

$$\bar{d}'_{kl} \geq 0, \; \forall\,(k,l) \in \bar{L}' \cup \bar{U}', \tag{3.102}$$

$$\underline{d}'_{kl} \geq 0, \; \forall\,(k,l) \in \bar{U}' \cup \bar{V}'. \tag{3.103}$$

Using (3.41) and (3.43), these conditions become

$$\pi_i - \pi_j + e_{ij} \leq 0, \; \forall\,(i,j) \in \bar{L}', \tag{3.104}$$

$$-f_{ij} \leq \pi_i - \pi_j + e_{ij} \leq 0, \; \forall\,(i,j) \in \bar{U}', \tag{3.105}$$

$$\pi_i - \pi_j + e_{ij} \leq -f_{ij}, \; \forall\,(i,j) \in \bar{V}'. \tag{3.106}$$

Let us further define the variables ρ_{ij} and δ_{ij} as follows:

$$\rho_{ij} = \begin{cases} 0 \;, & \forall\,(i,j) \in X' \cup \bar{L}', \\ \pi_j - \pi_i - e_{ij}, & \forall\,(i,j) \in \bar{U}' \cup \bar{V}', \\ f_{ij} \;, & \forall\,(i,j) \in Y', \end{cases} \tag{3.107}$$

$$\delta_{ij} = \begin{cases} \pi_j - \pi_i - e_{ij} - f_{ij}, & \forall\,(i,j) \in \bar{V}', \\ 0 \;, & \forall\,(i,j) \notin \bar{V}'. \end{cases} \tag{3.108}$$

It can be easily seen that π_j, ρ_{ij} and δ_{ij}, as defined above, give a feasible solution of the dual problem (3.92)-(3.95), which together with $\underline{x}_{ij}$ and $\underline{y}_{ij}$ satisfy the complementary slackness conditions (3.96)-(3.99). Hence $\underline{x}_{ij}$ and $\underline{y}_{ij}$ is an optimum solution of the problem (3.88)-(3.91) and $\underline{D} = D_{min}$.

3.3.3 Initial Optimum Basis Structure

Let D_{max} be the highest value of D' upto which the constraint (3.6) affects the optimum solution of the MCNCE problem. We now outline a procedure to obtain an optimum basis structure of the MCNCE problem for $D' = D_{max}$. Justification of the procedure is given subsequently.

Step 1: Solve the following minimum cost flow problem by any primal simplex method [10]:

$$\text{Minimize} \sum_{(i,j)\,\varepsilon\,A} c_{ij}\, x_{ij}\,, \tag{3.109}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = \begin{cases} -v\,, & \text{if } i=s\,, \\ 0\,, & \text{if } i\neq s,t,\ \forall\, i\ \varepsilon\ N, \\ v\,, & \text{if } i=t, \end{cases} \tag{3.110}$$

$$0 \leq x_{ij} \leq a_{ij}\,,\ \forall\,(i,j)\ \varepsilon\ A. \tag{3.111}$$

Let x^o_{ij} be the optimum solution. In this solution, let $B^o, \bar{L}^o, \bar{U}^o$ and $\bar{V}^o$ be the sets of basic arcs and nonbasic arcs having $x^o_{ij} = 0$, b_{ij} and a_{ij} respectively. The B^o consists of a spanning tree, $\bar{U}^o$ is empty and $\bar{L}^o \cup \bar{V}^o$ consists of nontree-arcs. Let $X^o = \{(i,j)\ \varepsilon\ B^o: 0 \leq x^o_{ij} \leq b_{ij}\}$ and $Y^o = \{(i,j)\ \varepsilon\ B^o: b_{ij} < x_{ij} \leq a_{ij}\}$.

Taking $X^o \cup Y^o$ as a basic tree, compute $\bar{c}^o_{kl}$ and $\underline{d}^o_{kl}$ using (3.40)-(3.43). Go to Step 2.

Step 2 : Let $\bar{G}^o \cup \underline{G}^o$ be the set of critical arcs. If $\bar{G}^o \cup \underline{G}^o$ is empty, go to step 4; otherwise select an arc $(p,q) \in \bar{G}^o \cup \underline{G}^o$ and go to step 3.

Step 3 : Let W_{pq} be the cycle formed when (p,q) is added to the basic tree. Depending upon the status of arc (p,q), define the orientation of W_{pq} as described in Section 3.3.1. Determine $\bar{w}$ using (3.53) and (3.54) and send additional flow $\bar{w}$ in W_{pq} along its orientation. Drop an arc $(\alpha,\beta) \in W_{pq}$ for which $\bar{w}_{\alpha\beta} = \bar{w}$. The new basic tree is $X^o \cup Y^o \cup \{(p,q)\} - \{(\alpha,\beta)\}$. Update $\bar{L}^o$, $\bar{U}^o$, $\bar{V}^o$ and recompute $\bar{c}^o_{kl}$, $\bar{d}^o_{kl}$, $\underline{c}^o_{kl}$ and $\underline{d}^o_{kl}$. Go to Step 2.

Step 4 : Let x_{ij} be the existing flow. Then $D_{max} = \sum_{(i,j)\in A} e_{ij}x_{ij} + \sum_{(i,j)\in A} f_{ij} \max.\{0, x_{ij}-b_{ij}\}$. If $D > D_{max}$, then x_{ij} is the optimum flow for the CNCE problem. Otherwise let $\bar{S}^o \cup \underline{S}^o$ be the set of active arcs. If $\bar{S}^o \cup \underline{S}^o$ is empty, then the CNCE problem is infeasible; otherwise define a number μ^o_{kl} for each $(k,l) \in \bar{S}^o \cup \underline{S}^o$ as

$$\mu^o_{kl} = \begin{cases} -\bar{c}^o_{kl} / \bar{d}^o_{kl}, & \text{if } (k,l) \in \bar{S}^o, \\ -\underline{c}^o_{kl} / \underline{d}^o_{kl}, & \text{if } (k,l) \in \underline{S}^o, \end{cases} \tag{3.112}$$

and select an arc $(g,h) \in \bar{S}^o \cup \underline{S}^o$ for which $\mu^o_{gh} = \min_{(k,l)\in \bar{S}^o \cup \underline{S}^o} \{\mu^o_{kl}\}$.

The initial optimum basis structure for $D' = D_{max}$ is $(X^o \cup Y^o \cup \{(g,h)\}, L^o, U^o, V^o)$, where L^o, U^o and V^o are the sets of arcs in $\bar{L}^o \cup \bar{U}^o \cup \bar{V}^o - \{(g,h)\}$ having flow equal to 0, b_{ij} and a_{ij} respectively.

In Step 1, an optimum solution of the MCNCE problem without the constraint (3.6) is obtained. According to our notations, this solution satisfies the following conditions:

$$\bar{c}^o_{kl} \geq 0, \ \forall (k,l) \in \bar{L}^o \cup \bar{U}^o \quad \text{and} \quad \underline{c}^o_{kl} \geq 0, \ \forall (k,l) \in \bar{U}^o \cup \bar{V}^o. \tag{3.113}$$

In Steps 2 and 3, an optimum solution of (3.109)-(3.111) requiring minimum value of D' is obtained. This is done by selecting an arc $(p,q) \in \bar{G}^o$ for which $\bar{c}^o_{pq} = 0$ and $\bar{d}^o_{pq} < 0$, or an arc $(p,q) \in \underline{G}^o$ for which $\underline{c}^o_{pq} = 0$ and $\underline{d}^o_{pq} < 0$, and performing the primal simplex pivot iteration. In view of our interpretation for $\bar{c}^o_{pq}$, $\bar{d}^o_{pq}$, $\underline{c}^o_{pq}$ and $\underline{d}^o_{pq}$, the value of Z does not change but the value of D' decreases by the pivot iteration. Since $\bar{c}^o_{pq} = \underline{c}^o_{pq} = 0$, the conditions in (3.113) are maintained. At the end of Step 2, a solution is obtained in which D' can not be decreased without increasing Z. Hence this value of D' is the value of D_{max}. It is obvious that if $D > D_{max}$, this solution optimizes the CNCE problem.

The optimum basis structure for $D' = D_{max}$ is constructed in Step 4. If the set of active arcs is empty, then it follows from Theorem 3.6 that the MCNCE problem is infeasible for all $D' < D_{max}$. Hence the CNCE problem is infeasible if $D < D_{max}$. If the set of active arcs is nonempty, then an active arc which yields the minimum value of the pivot ratio is selected as the pivot arc. We now show that this

From (3.45) we get $\mu^o = \mu^o_{gh}$. Consider any arc $(k,l) \varepsilon \bar{S}^o \cup \underline{S}^o$. Clearly

$$\mu^o = \mu^o_{gh} \leq \mu^o_{kl} = \begin{cases} -\bar{c}^o_{kl} / \bar{d}^o_{kl}, & \text{if } (k,l) \varepsilon \bar{S}^o, \\ -\underline{c}^o_{kl} / \underline{d}^o_{kl}, & \text{if } (k,l) \varepsilon \underline{S}^o. \end{cases} \tag{3.114}$$

After rearrangement (3.114) reduces to (3.46) and (3.47). Now consider any arc $(k,l) \not\varepsilon \bar{S}^o \cup \underline{S}^o$. Termination of Step 2 guarentees that there is no critical arc. Hence (k,l) is a passive arc. This implies that

$$\bar{d}^o_{kl} \geq 0, \text{ if } (k,l) \varepsilon \bar{L}^o \cup \bar{U}^o; \text{ and } \underline{d}^o_{kl} \geq 0, \text{ if } (k,l) \varepsilon \bar{U}^o \cup \bar{V}^o. \tag{3.115}$$

In view of (3.113) and (3.115), the optimality conditions (3.46) and (3.47) are satisfied by the arc (k,l).

3.3.4 Physical Interpretation

The algorithm first obtains an optimum solution of the MCNCE problem for a large value of D'. This value of D' is then continuously decreased and optimality is maintained. At each step, the algorithm keeps a basic tree and the pivot arc is used to decrease D' by performing a simplex-like iteration. The value of Z increases continuously and the pivot ratio indicates the rate of increase in Z per unit decrease in D'. The algorithm proceeds in this manner until $D' = D$, when the optimum solution of the MCNCE problem is the optimum solution of the CNCE problem.

The algorithm is essentially based on the trade-off of D' with Z. The pivot arc plays the central role in this trade-off. The pivot

arc is selected among the nontree-arcs which are classified into three categories, viz., critical arcs, active arcs and passive arcs. Critical arcs are those nontree-arcs which may lead to decrease in D' without increasing Z if any one of those arcs becomes the pivot arc. Active arcs may also lead to decrease in D' but by increasing Z. Unlike critical and active arcs, passive arcs do not lead to decrease in D'. While obtaining the initial optimum basis structure, critical arcs are used to decrease D'. Since critical arcs do not appear in later steps, active arcs are then used to decrease D'. If there is no active arc, the value of D' can not be decreased and the algorithm terminates. However, if there are active arcs, then an active arc which yields minimum value of the pivot ratio is selected as the pivot arc. This selection criteria is quite intuitive as it implies that at every step D' is decreased by incurring minimum increase in Z. In subsequent iterations, value of the pivot ratio increases indicating that decreasing D' becomes costlier.

3.4 *DESCRIPTION OF THE ALGORITHM*

A formal description of the CNCE algorithm is given below.

<u>Step 1</u> : Obtain the initial optimum basis structure $(X \cup Y \cup \{(p,q)\}, L, U, V)$ for $D' = D_{max}$ by the procedure described in Section 3.3.3. Let x_{ij} be the optimum flow. Set $\bar{Z} = \sum_{(i,j) \in A} c_{ij} x_{ij}$ and $\bar{D} = D_{max}$. If $D \geq \bar{D}$, go to Step 5; otherwise go to Step 2.

Step 2 : Let W_{pq} be the cycle consisting of basic arcs. Define the orientation of W_{pq} along (p,q) if $x_{pq} = 0$ and (p,q) is an unsaturated basic arc, or $x_{pq} = b_{pq}$ and (p,q) is a saturated basic arc. Otherwise define the orientation of W_{pq} opposite to (p,q). Let $\bar{W}_{pq}$ and $\underline{W}_{pq}$ be the sets of arcs in W_{pq} along and opposite to the orientation of W_{pq} respectively. Define a number $\bar{w}_{ij}$ for each $(i,j) \in W_{pq}$ as follows:

$$\bar{w}_{ij} = \begin{cases} b_{ij} - x_{ij} & , \text{ if } (i,j) \in \bar{X} \cap \bar{W}_{pq} , \\ x_{ij} & , \text{ if } (i,j) \in \bar{X} \cap \underline{W}_{pq} , \\ a_{ij} - x_{ij} & , \text{ if } (i,j) \in \bar{Y} \cap \bar{W}_{pq} , \\ x_{ij} - b_{ij} & , \text{ if } (i,j) \in \bar{Y} \cap \underline{W}_{pq} , \end{cases}$$

where $\bar{X}$ and $\bar{Y}$ are the sets of unsaturated and saturated basic arcs respectively.

Compute

$$\bar{w} = \min_{(i,j) \in W_{pq}} \{ \bar{w}_{ij} \} ,$$

and

$$\bar{r} = \min. \{ \bar{w}, (\bar{D} - D) / \bar{d}_{pq} \} .$$

Update x_{ij}, $\bar{Z}$ and $\bar{D}$ as follows:

$$x_{ij} = \begin{cases} x_{ij} + \bar{r} & , \forall (i,j) \in \bar{W}_{pq} , \\ x_{ij} - \bar{r} & , \forall (i,j) \in \underline{W}_{pq} , \\ x_{ij} & , \forall (i,j) \notin W_{pq} , \end{cases}$$

$$\bar{Z} = \begin{cases} \bar{Z} + \bar{r}\,\bar{c}_{pq}, & \text{if } (p,q) \text{ is an unsaturated basic arc,} \\ \bar{Z} + \bar{r}\,\underline{c}_{pq}, & \text{if } (p,q) \text{ is a saturated basic arc,} \end{cases}$$

$$\bar{D} = \begin{cases} \bar{D} + \bar{r}\,\bar{d}_{pq}, & \text{if } (p,q) \text{ is an unsaturated basic arc,} \\ \bar{D} + \bar{r}\,\underline{d}_{pq}, & \text{if } (p,q) \text{ is a saturated basic arc.} \end{cases}$$

If $\bar{D} = D$, go to Step 5; otherwise go to Step 3.

Step 3 : Identify an arc $(\alpha,\beta) \in W_{pq}$ for which $\bar{w}_{\alpha\beta} = \bar{w}$. The new basic tree is $X \cup Y \cup \{(p,q)\} - \{(\alpha,\beta)\}$. Recompute $\bar{c}_{kl}$, $\bar{d}_{kl}$, $\underline{c}_{kl}$ and $\underline{d}_{kl}$. Let $\bar{S} \cup \underline{S}$ be the set of active arcs. If $\bar{S} \cup \underline{S}$ is empty, go to Step 4; otherwise define a number μ_{kl} for each $(k,l) \in \bar{S} \cup \underline{S}$ as follows:

$$\mu_{kl} = \begin{cases} -\bar{c}_{kl} / \bar{d}_{kl}, & \text{if } (k,l) \in \bar{S}, \\ -\underline{c}_{kl} / \underline{d}_{kl}, & \text{if } (k,l) \in \underline{S}, \end{cases}$$

and select an arc $(g,h) \in \bar{S} \cup \underline{S}$ for which $\mu_{gh} = \min_{(k,l) \in \bar{S} \cup \underline{S}} \{\mu_{kl}\}$. Set $(p,q) = (g,h)$, which is an unsaturated basic arc if $(p,q) \in \bar{S}$, and a saturated basic arc if $(p,q) \in \underline{S}$. Update the basis structure and go to Step 2.

Step 4 : The CNCE problem is infeasible. STOP.

Step 5 : The solution x_{ij} and $y_{ij} = \max.\{0, x_{ij} - b_{ij}\}$ is the optimum solution of the CNCE problem with the value of objective function $\bar{Z}$. STOP.

3.4.1 The Tradeoff Curve

The CNCE algorithm can be used to solve the MCNCE problem for all feasible values of D'. If the optimum objective function value of the MCNCE problem is plotted against D', a piecewise linear convex function is obtained. In this curve, the slope of a linear segment is negative of the value of the pivot ratio in the associated optimum basis structure. The slope of this curve is zero for $D' > D_{max}$. The slope decreases as D decreases and finally becomes $-\infty$ for $D = D_{min}$. Since this curve depicts the tradeoff between D' and Z, we refer to it as the tradeoff curve.

3.4.2 Near-Optimum Integer Solution

The optimum solutions of the MCNCE problem obtained by the algorithm at the end points of the characteristic intervals are integral. If the prescribed value of D is not one of these end points, the optimum solution of the CNCE problem provided by the algorithm may be nonintegral. In such cases, the following modification in the algorithm yields a near-optimum integer solution. In the last iteration, when the prescribed value of D is met exactly, $\bar{r}$ may be less than $\bar{w}$ and may be noninteger. Then, incrementing $\bar{r}$ to the nearest integer, i.e., $[\bar{r} + 1]$, yields an integral as well as a feasible flow. The difference in the objective function values of this integral solution and the optimum solution is $([\bar{r} + 1] - \bar{r})\,\bar{c}_{pq}$ if (p,q) is an unsaturated basic arc and $([\bar{r} + 1] - \bar{r})\,\underline{c}_{pq}$ if (p,q) is a saturated basic arc. If the value of $\bar{c}_{pq}$ or $\underline{c}_{pq}$ is small, the integral solution is near-optimum.

3.5 NUMERICAL EXAMPLE

In this section, a numerical example is solved to illustrate various steps of the CNCE algorithm. The network is shown in Fig. 3.1. The data associated with arcs is given in Table 3.1. The nodes 1 and 6 respectively represent source and sink. The flow to be transshipped from source to sink is 12 units. It is desired to obtain the tradeoff curve of the numerical example.

Table 3.1 : Data of the numerical example

(i,j)	a_{ij}	b_{ij}	c_{ij}	e_{ij}	f_{ij}
(1,2)	12	6	2	0	1
(1,3)	8	8	3	0	0
(2,3)	5	3	2	0	1
(2,4)	10	5	4	0	2
(2,5)	∞	5	1	0	3
(3,5)	10	9	2	0	4
(4,6)	12	8	2	0	3
(5,4)	5	4	5	0	2
(5,6)	15	6	2	0	2

The CNCE algorithm obtains the tradeoff curve in six iterations. The computations involved in these iterations are summarized in Table 3.2. In the table, the term 'status' indicates the set to which an arc belongs. The symbol ↑ denotes the pivot arc and ↓ denotes the basic arc leaving the basis. Basis in various iterations are shown in Fig. 3.2. In this figure, the pivot arcs are indicated by dashed lines and saturated basic arcs are drawn in double lines. The tradeoff curve is depicted in Fig. 3.3.

3.6 RELATED NETWORK PROBLEMS

In this section, we consider (i) a variation of the CNCE problem; and (ii) a bicriteria network problem. We show how the CNCE algorithm can be used to solve these problems.

3.6.1 A Variation of the CNCE Problem

In the application of the CNCE problem cited in Section 3.1, our objective was to minimize the total time of transportation subject to the consideration that the cost of transportation and capacity expansion does not exceed a prescribed budget D. Our objective could as well be to minimize the cost of transportation and capacity expansion subject to the consideration that the total time of transportation does not exceed a prescribed value. Problems of this kind can be formulated as follows:

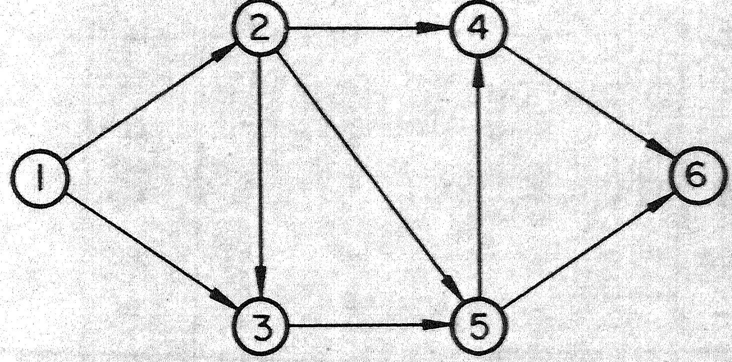

Fig. 3.1 Numerical example for the CNCE problem

Table 3.2 : Solution of the numerical example

Iteration No.	$\bar{Z}$	$\bar{D}$	μ	$\bar{w}$		Tree-arcs					Nontree-arcs			
					(k,l)	(1,3)	(1,2)	(2,5)	(5,6)	(4,6)	(2,4)	(2,3)	(3,5)	(5,4)
					x_{kl}	0	12	12	12	0	0	0	0	0
					$\bar{c}_{kl}$	-	-	-	-	-	3	1	2	5
					$\bar{d}_{kl}$	-	-	-	-	-	-5	1	-4	-2
1.	60	39	0.5	6	$\underline{c}_{kl}$	-	-	-	-	-	-	-	-	-
					$\underline{d}_{kl}$	-	-	-	-	-	-	-	-	-
					Status	X	Y	Y	Y	X	$\bar{S}$	H	$\bar{S}$	$\bar{S}$
					μ_{kl}	-	-	-	-	-	0.6	-	0.5	2.5
					$\bar{w}_{kl}$	8	6	7	-	-	-	-	9	-
							↓						↑	
					(k,l)	(1,3)	(3,5)	(2,5)	(5,6)	(4,6)	(2,4)	(2,3)	(1,2)	(5,4)
					x_{kl}	6	6	6	12	0	0	0	6	0
					$\bar{c}_{kl}$	-	-	-	-	-	3	3	2	5
					$\bar{d}_{kl}$	-	-	-	-	-	-5	-3	-3	-2
2.	72	15	0.6	1	$\underline{c}_{kl}$	-	-	-	-	-	-	-	-2	-
					$\underline{d}_{kl}$	-	-	-	-	-	-	-	2	-
					Status	X	X	Y	Y	X	$\bar{S}$	$\bar{S}$	H	$\bar{S}$
					μ_{kl}	-	-	-	-	-	0.6	1.0	-	2.5
					$\bar{w}_{kl}$	-	-	1	6	8	5	-	-	-
								↓			↑			
					(k,l)	(1,3)	(3,5)	(2,4)	(5,6)	(4,6)	(2,5)	(2,3)	(1,2)	(5,4)
					x_{kl}	6	6	1	11	1	5	0	6	0
					$\bar{c}_{kl}$	-	-	-	-	-	3	0	-1	5
					$\bar{d}_{kl}$	-	-	-	-	-	-2	2	2	-2
3.	75	10	1.0	4	$\underline{c}_{kl}$	-	-	-	-	-	-3	-	1	-
					$\underline{d}_{kl}$	-	-	-	-	-	5	-	-1	-
					Status	X	X	X	Y	X	$\bar{S}$	H	$\underline{S}$	$\bar{S}$
					μ_{kl}	-	-	-	-	-	1.5	-	1.0	2.5
					$\bar{w}_{kl}$	6	6	4	5	7	-	-	6	-
								↓					↑	

	(k,l)	(1,3)	(3,5)	(1,2)	(5,6)	(4,6)	(2,5)	(2,3)	(2,4)	(5,4)
	x_{kl}	2	2	10	7	5	5	0	5	0
	$\bar{c}_{kl}$	–	–	–	–	–	2	1	-1	5
	$\bar{d}_{kl}$	–	–	–	–	–	-1	1	1	-2
4. 79 6 2.0 4	$\underline{c}_{kl}$	–	–	–	–	–	-2	–	1	–
	$\underline{d}_{kl}$	–	–	–	–	–	4	–	1	–
	Status	X	X	Y	Y	X	$\bar{S}$	H	H	$\bar{S}$
	μ_{kl}	–	–	–	–	–	2	–	–	2.5
	$\bar{w}_{kl}$	6	7	4	–	–	5	–	–	–
				↓			↑			
	(k,l)	(1,3)	(3,5)	(2,5)	(5,6)	(4,6)	(1,2)	(2,3)	(2,4)	(5,4)
	x_{kl}	6	6	1	7	5	6	0	5	0
	$\bar{c}_{kl}$	–	–	–	–	–	2	3	-3	5
	$\bar{d}_{kl}$	–	–	–	–	–	0	0	2	-2
5. 87 2 2.5 1	$\underline{c}_{kl}$	–	–	–	–	–	-2	–	3	–
	$\underline{d}_{kl}$	–	–	–	–	–	1	–	0	–
	Status	X	X	X	Y	X	H	H	H	$\bar{S}$
	μ_{kl}	–	–	–	–	–	–	–	–	2.5
	$\bar{w}_{kl}$	–	–	–	1	3	–	–	–	4
					↓					↑
	(k,l)	(1,3)	(3,5)	(2,5)	(5,4)	(4,6)	(1,2)	(2,3)	(2,4)	(5,6)
	x_{kl}	6	6	1	1	6	6	0	5	6
	$\bar{c}_{kl}$	–	–	–	–	–	2	3	2	5
6. 92 0 – –	$\bar{d}_{kl}$	–	–	–	–	–	0	0	0	0
	$\underline{c}_{kl}$	–	–	–	–	–	-2	–	-2	-5
	$\underline{d}_{kl}$	–	–	–	–	–	1	–	2	3
	Status	X	X	X	X	X	H	H	H	H

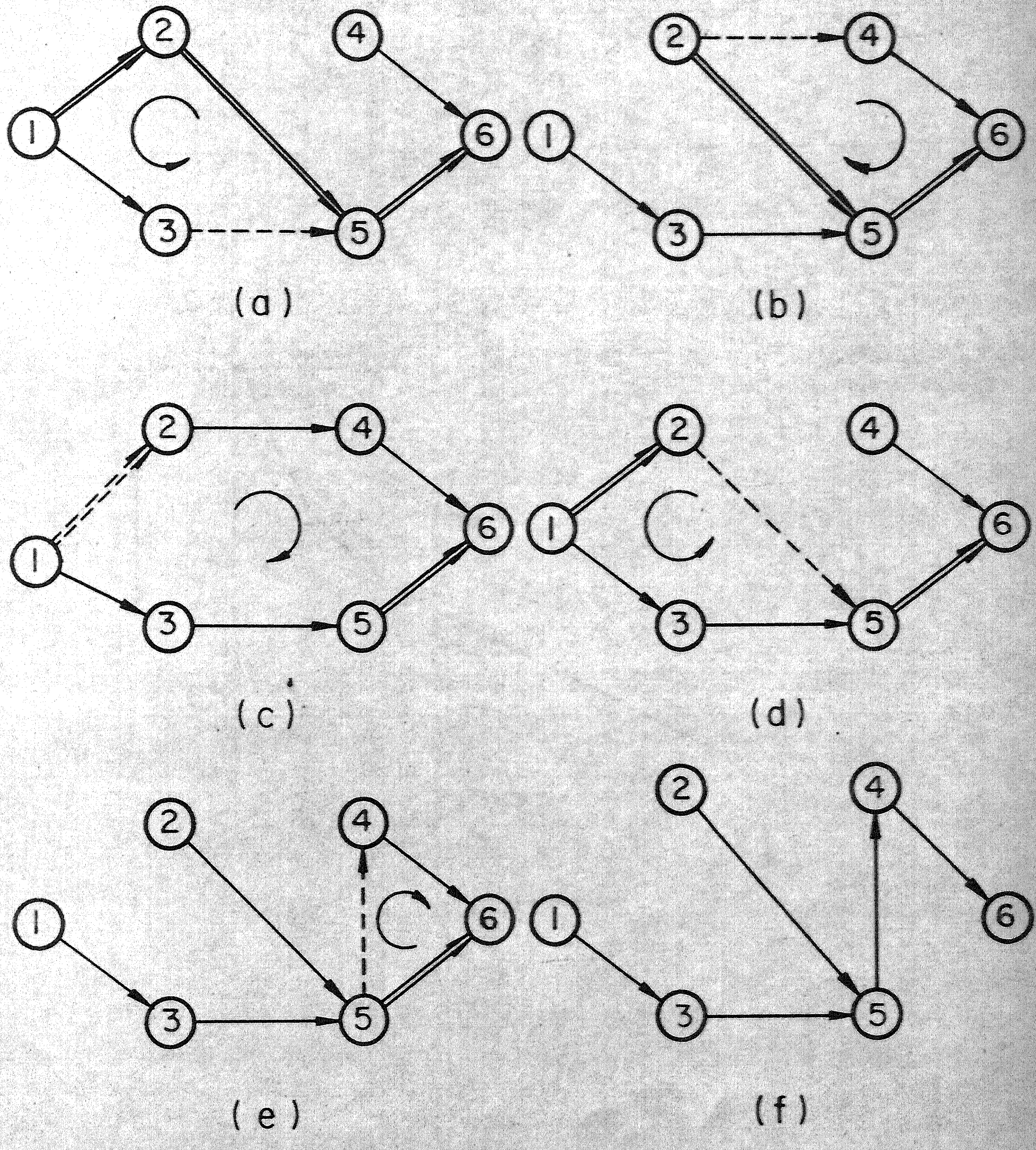

Fig. 3.2 Basis in various iterations

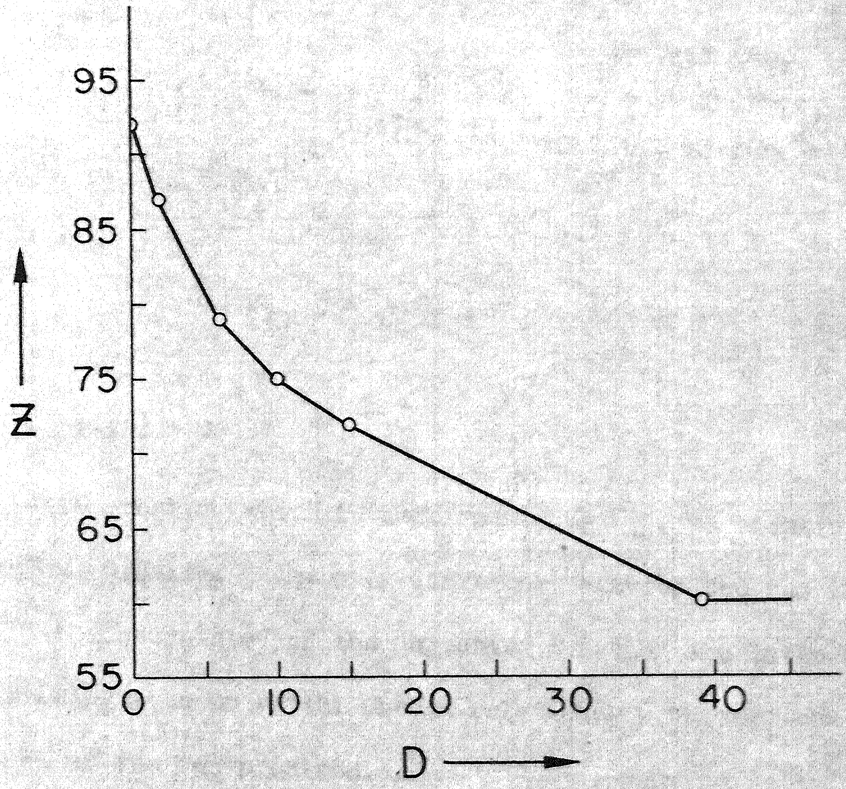

Fig. 3.3 The tradeoff curve

$$\text{Minimize} \quad \sum_{(i,j)\,\varepsilon\, A} e_{ij}x_{ij} + \sum_{(i,j)\,\varepsilon\, A} f_{ij}y_{ij}\,, \tag{3.116}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = \begin{cases} -v\,, & \text{if } i=s, \\ 0\,, & \text{if } i \neq s,t, \quad \forall\, i \,\varepsilon\, N, \\ v\,, & \text{if } i=t, \end{cases} \tag{3.117}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij}, \ \forall\, (i,j) \ \varepsilon\ A, \tag{3.118}$$

$$0 \leq y_{ij} \leq a_{ij} - b_{ij}, \ \forall\, (i,j) \ \varepsilon\ A, \tag{3.119}$$

$$\sum_{(i,j)\,\varepsilon\, A} c_{ij}\, x_{ij} \leq Z. \tag{3.120}$$

This problem, which is referred to as P1, is closely related to the CNCE problem. The only difference between the two is that one of the constraints and the objective function are interchanged. The following theorem establishes a relationship between the optimum solutions of the two problems.

Theorem 3.7 : Let x^o_{ij} be the optimum solution of the CNCE problem for $D = D^o \leq D_{max}$ with Z^o as the value of objective function. Then, x^o_{ij} is the optimum solution of P1 for $Z = Z^o$ with D^o as the value of objective function.

Proof : Suppose that the theorem is not true and there exists a solution x^*_{ij} of P1 for $Z = Z^o$ with the value of objective function $D^* < D^o$. Let $Z^* = \sum_{(i,j)\,\varepsilon\, A} c_{ij}\, x^*_{ij}$. We note that $Z^o = Z^*$ because x^*_{ij} is also a feasible solution of the CNCE problem for $D = D^o$ and $Z^* < Z^o$ contradicts the optimality of the solution x^o_{ij}.

Further, $D^* < D^o$ implies that in the optimum solution of the CNCE problem for $D = D^o$, the constraint (3.5) is not a binding constraint and, hence, $\mu = 0$. This contradicts our assumption that $\mu > 0$ for $D \leq D_{max}$. Hence $Z^* = Z^o$, $D^* = D^o$ and x^o_{ij} is the optimum solution of P1 for $Z = Z^o$.

It is now obvious that the CNCE algorithm solves P1. The CNCE algorithm can also be viewed as if it treats Z as a parameter and increases its value until either the prescribed value of Z is reached or infeasibility of P1 is indicated.

3.6.2 A Bicriteria Network Problem

Consider the following bicriteria network problem:

$$\text{VMIN} \begin{Bmatrix} \sum_{(i,j) \varepsilon A} c_{ij} x_{ij} \\ \sum_{(i,j) \varepsilon A} e_{ij} x_{ij} + \sum_{(i,j) \varepsilon A} f_{ij} y_{ij} \end{Bmatrix} \tag{3.121}$$

subject to

$$\sum_{(j,i) \varepsilon I(i)} x_{ji} - \sum_{(i,j) \varepsilon O(i)} x_{ij} = \begin{cases} -v \text{ , if } i=s, \\ 0 \text{ , if } i \neq s,t, \\ v \text{ , if } i=t, \end{cases} \forall\, i \,\varepsilon\, N, \tag{3.122}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij}, \; \forall\, (i,j) \,\varepsilon\, A, \tag{3.123}$$

$$0 \leq y_{ij} \leq a_{ij} - b_{ij}, \; \forall\, (i,j) \,\varepsilon\, A. \tag{3.124}$$

The parametric programming techniques have been used extensively to solve multicriteria mathematical programs. For bicriteria mathematical programs, Bacopoulus and Singer [4] have suggested the 'constraint criteria approach', i.e., maximizing (or minimizing) one

criteria and keeping the other criteria as a constraint. They showed how all the efficient solutions can be generated by parametrically varying the level of the constraint.

The CNCE algorithm treats the criteria ($\sum_{(i,j) \varepsilon A} e_{ij} x_{ij} + \sum_{(i,j) \varepsilon A} f_{ij} y_{ij}$) as a constraint and varies its level parametrically. Thus the solutions of the CNCE problem, which generate the tradeoff curve, are the efficient solutions of (3.121) - (3.124) and the points, where slope of the curve changes, correspond to extreme efficient solutions.

3.7 SPECIAL CASES

In this section, we consider some special cases of the CNCE problem and point out the resulting simplifications in the CNCE algorithm.

3.7.1 Constrained Minimum Cost Flow Problem

The mathematical statement of the Constrained Minimum Cost Flow (CMCF) problem is as follows:

$$\text{Minimize} \sum_{(i,j)\ \varepsilon\ A} c_{ij}\, x_{ij}\,, \tag{3.125}$$

subject to

$$\sum_{(j,i)\ \varepsilon\ I(i)} x_{ji} - \sum_{(i,j)\ \varepsilon\ O(i)} x_{ij} = \begin{cases} -v\,, & \text{if } i=s, \\ 0\,, & \text{if } i \neq s,t, \quad \forall\, i \,\varepsilon\, N, \\ v\,, & \text{if } i=t, \end{cases} \tag{3.126}$$

$$0 \leq x_{ij} \leq b_{ij} \; , \; \forall \, (i,j) \; \varepsilon \; A, \tag{3.127}$$

$$\sum_{(i,j) \, \varepsilon A} e_{ij} \; x_{ij} \leq D. \tag{3.128}$$

In the CNCE problem, if we put $f_{ij} = 0$ and $a_{ij} = b_{ij}$ for each $(i,j) \; \varepsilon \; A$, then it reduces to the CMCF problem. Following simplifications result in the CNCE algorithm when it is applied to the CMCF problem: The sets of saturated basic arcs $(\bar{Y})$ and the nonbasic arcs at their upper bounds (V) are empty. The optimum basis structure of the CMCF problem is $(X \cup \{(p,q)\}, L, U)$ where (p,q) is always an unsaturated basic arc. Only two numbers, $\bar{c}_{kl}$ and $\bar{d}_{kl}$, need to be defined for each nontree-arc (k,l). Hence the sets $\underline{S}$ and $\underline{G}$ are empty. Further, the optimality conditions for the CMCF problem are

$$\bar{c}_{pq} + \mu \, \bar{d}_{pq} = 0, \tag{3.129}$$

$$\bar{c}_{kl} + \mu \, \bar{d}_{kl} \geq 0, \quad (k,l) \; \varepsilon \; L \cup U. \tag{3.130}$$

On account of these simplifications, obtaining the characteristic interval and performing the dual simplex iteration is easier. However, interpretation of the algorithm does not change.

3.7.2 Bicriteria Transportation Problem

The Bicriteria Transportation (BT) problem has been considered by Aneja and Nair [3]. They suggested a method, which obtains all the efficient solutions of the BT problem by solving a sequence of transportation problems. In the CNCE problem, if we put $a_{ij} = b_{ij} = \infty$ and $f_{ij} = 0$ for each $(i,j) \; \varepsilon \; A$, and the underlying network is that of a

transportation problem, then the CNCE algorithm obtains all the efficient solutions of the BT problem. The CNCE algorithm, then, solves one transportation problem in the beginning and in each subsequent iteration an effort equivalent to that of performing one simplex iteration is required. Hence our algorithm is computationally superior to the method of Aneja and Nair [3] for solving the BT problem.

3.7.3 Constrained Shortest Path Problem

The Constrained Shortest Path (CSP) problem is to determine the shortest path from source to sink satisfying an additional linear constraint. The CSP problem can be formulated as the CMCF problem with additional integrality requirements on variables. Recently, the CSP problem is proved to be NP-complete by showing that the knapsack problem is a special case of the CSP problem [49]. Since the CSP problem is NP-complete, efficient algorithms yielding near-optimum solutions are highly desirable. The CMCF algorithm incorporating the integrality modification suggested in Section 3.4.2 provides such a solution.

3.7.4 Other Special Cases

The problem to allocate a given budget D to increase the capacities of various arcs so that the cost of flow in the network is minimized, can be formulated as follows:

$$\text{Minimize } Z = \sum_{(i,j) \varepsilon A} c_{ij}\, x_{ij}\,, \tag{3.131}$$

subject to

$$\sum_{(j,i) \varepsilon I(i)} x_{ji} - \sum_{(i,j) \varepsilon O(i)} x_{ij} = \begin{cases} -v\,, & \text{if } i=s, \\ 0\,, & \text{if } i \neq s,t, \quad \forall\, i \,\varepsilon\, N, \\ v\,, & \text{if } i=t, \end{cases} \tag{3.132}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij}, \ \forall\, (i,j) \ \varepsilon \ A, \tag{3.133}$$

$$0 \leq y_{ij} \leq a_{ij} - b_{ij}, \ \forall\, (i,j) \ \varepsilon \ A, \tag{3.134}$$

$$\sum_{(i,j) \varepsilon A} f_{ij}\, y_{ij} \leq D. \tag{3.135}$$

We refer to the problem (3.131)-(3.135) as the Capacity Expansion to Minimize Cost of Flow (CEMCF) problem. In the CNCE problem, if we put $e_{ij} = 0$ for each $(i,j) \ \varepsilon \ A$, it reduces to the CEMCF problem. There is no change in the CNCE algorithm when it is applied to the CEMCF problem.

Consider a variation of the CEMCF problem in which no fixed budget is prescribed and capacities of arcs can be increased as long as the total cost of flow and capacity expansion is minimized. The formulation of this problem can be given by (3.88)-(3.91). The solution provided by the CNCE algorithm, when the set of active arcs is empty, is the optimum solution of this problem.

The problem of allocating a prescribed budget D to increase capacities of various arcs so that the flow in the network is maximized, is a special case of the CEMCF problem. This problem can be solved by the CNCE algorithm but the problem is too simple to be solved by the

CNCE algorithm. Instead, we consider this problem as a special case of the parametric network feasibility problem considered in the next section. Similar comments apply to the constrained maximum flow problem which is a special case of the CMCF problem. We consider the constrained maximum flow problem in Chapter IV.

3.8 *PARAMETRIC NETWORK CAPACITY EXPANSION PROBLEM*

In this section, we consider capacitated transshipment networks in which supplies at source nodes and demands at sink nodes are linear functions of a single parameter, and the capacities of arcs can be increased by incurring additional cost. The problem considered is to determine the least cost of increasing arc capacities so that the changing demands are met by the changing supplies, resulting from the changes in λ. We term this problem as the <u>Parametric Network Capacity Expansion</u> (PNCE) problem. The mathematical statement of the PNCE problem is as follows:

$$\text{Minimize} \quad Z(\lambda) = \sum_{(i,j)\,\varepsilon\, A} c_{ij} y_{ij}\,, \tag{3.136}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = r_i^0 + \lambda r_i^*, \quad \forall\; i\; \varepsilon\; N, \tag{3.137}$$

$$0 \le x_{ij} \le b_{ij} + y_{ij}, \quad \forall\; (i,j)\; \varepsilon\; A, \tag{3.138}$$

$$0 \le y_{ij} \le a_{ij} - b_{ij}, \quad \forall\; (i,j)\; \varepsilon\; A, \tag{3.139}$$

where $\quad c_{ij} > 0$ for each $(i,j)\ \varepsilon\ A.$

The PNCE problem is a blend of the parametric network feasibility problem considered in Chapter II and the capacity expansion problems considered in this chapter. We use the concepts developed in both of these chapters to obtain an efficient algorithm to solve the PNCE problem for all values of λ for which a feasible flow exists.

3.8.1 Optimality Conditions

Let us partition A into the sets X, Y, L, U and V such that $X \cup Y$ is a spanning tree. The optimum basis structure of the PNCE problem is defined as $(X \cup Y, L, U, V)$ which is similar to the optimum basis structure of the MCNCE problem. The only difference between the two is that the basis of the PNCE problem consists of a spanning tree instead of a spanning tree augmented by a nontree-arc for the MCNCE problem. This difference arises because the additional linear constraint is absent in the PNCE problem. We now derive the necessary and sufficient conditions to be satisfied by the optimum basis structure of the PNCE problem. The dual of the PNCE problem is

$$\text{Maximize} \sum_{i \in N} (r_i^o + \lambda r_i^*)\pi_i - \sum_{(i,j) \in A} b_{ij}\rho_{ij} - \sum_{(i,j) \in A} (a_{ij} - b_{ij})\delta_{ij},$$

subject to $\qquad (3.140)$

$$\pi_j - \pi_i - \rho_{ij} \leq 0 \;, \;\forall (i,j) \in A, \qquad (3.141)$$

$$\rho_{ij} - \delta_{ij} \leq c_{ij}, \;\forall (i,j) \in A, \qquad (3.142)$$

$$\rho_{ij} \geq 0 \text{ and } \delta_{ij} \geq 0, \;\forall (i,j) \in A, \qquad (3.143)$$

where π_i, ρ_{ij} and δ_{ij} are the dual variables associated with the constraints (3.137), (3.138) and (3.139) respectively.

The complementary slackness conditions for the PNCE problem are

$$\rho_{ij}(b_{ij}+y_{ij}-x_{ij}) = 0 \quad , \forall (i,j) \in A, \tag{3.144}$$

$$\delta_{ij}(a_{ij}-b_{ij}-y_{ij}) = 0 \quad , \forall (i,j) \in A, \tag{3.145}$$

$$x_{ij}(\rho_{ij}+\pi_i-\pi_j) = 0 \quad , \forall (i,j) \in A, \tag{3.146}$$

$$y_{ij}(c_{ij}+\delta_{ij}-\rho_{ij}) = 0 \quad , \forall (i,j) \in A. \tag{3.147}$$

The following theorem, which is similar to Theorem 3.1, is used to derive an alternate representation of the complementary slackness conditions.

<u>Theorem 3.8</u> : Every optimum solution of the PNCE problem satisfies

$$y_{ij}(b_{ij}+y_{ij}-x_{ij}) = 0 \; , \forall (i,j) \in A. \tag{3.148}$$

<u>Proof</u> : Suppose $y_{ij} > 0$ and $x_{ij} < b_{ij}+y_{ij}$ for an arc (i,j). It follows from (3.144) and (3.147) that

$$x_{ij} < b_{ij} + y_{ij} \Longrightarrow \rho_{ij} = 0 \, , \tag{3.149}$$

$$y_{ij} > 0 \qquad \Longrightarrow c_{ij}+\delta_{ij}-\rho_{ij} = 0. \tag{3.150}$$

Hence

$$x_{ij} < b_{ij}+y_{ij} \text{ and } y_{ij} > 0 \Longrightarrow c_{ij}+\delta_{ij} = 0. \tag{3.151}$$

Since $\delta_{ij} \geq 0$, (3.151) implies that $c_{ij} \leq 0$ which contradicts our assumption that $c_{ij} > 0$.

Using Theorem 3.8, the conditions (3.144)-(3.147) can be shown equivalent to the following conditions:

$$0 < x_{ij} < b_{ij} \implies \pi_j - \pi_i = 0 \text{ and } y_{ij} = 0, \ \forall\, (i,j) \in A, \tag{3.152}$$

$$b_{ij} < x_{ij} < a_{ij} \implies \pi_j - \pi_i = c_{ij} \text{ and } y_{ij} = x_{ij} - b_{ij}, \ \forall\, (i,j) \in A, \tag{3.153}$$

$$\pi_j - \pi_i < 0 \implies x_{ij} = 0 \text{ and } y_{ij} = 0, \ \forall\, (i,j) \in A, \tag{3.154}$$

$$0 < \pi_j - \pi_i < c_{ij} \implies x_{ij} = b_{ij} \text{ and } y_{ij} = 0, \ \forall\, (i,j) \in A, \tag{3.555}$$

$$\pi_j - \pi_i > c_{ij} \implies x_{ij} = a_{ij} \text{ and } y_{ij} = a_{ij} - b_{ij}, \ \forall\, (i,j) \in A. \tag{3.156}$$

It is seen from these conditions that in the optimum solution $y_{ij} = \max. \{0, x_{ij} - b_{ij}\}$ for each $(i,j) \in A$. This relationship permits us to consider y_{ij} implicitly. It further follows from (3.152)-(3.156) that the necessary and sufficient conditions for a feasible basis structure of the PNCE problem to be an optimum basis structure are that there exist numbers π_j satisfying the following conditions:

$$\pi_j - \pi_i = 0, \ \forall\, (i,j) \in X, \tag{3.157}$$

$$\pi_j - \pi_i = c_{ij}, \forall\, (i,j) \in Y, \tag{3.158}$$

$$\pi_j - \pi_i \le 0, \forall\, (i,j) \in L, \tag{3.159}$$

$$0 \le \pi_j - \pi_i \le c_{ij}, \forall\, (i,j) \in U, \tag{3.160}$$

$$\pi_j - \pi_i \ge c_{ij}, \forall\, (i,j) \in V. \tag{3.161}$$

We refer to the conditions (3.157)-(3.161) as the <u>optimality conditions</u> for the PNCE problem. In the subsequent discussion in this section, the optimum basis structure and the optimality conditions

are used in reference to the PNCE problem. We refer to π_j satisfying the optimality conditions as the <u>optimum dual variables</u> associated with the optimum basis structure $(X \cup Y, L, U, V)$. It may be noted that given any feasible basis structure $(X \cup Y, L, U, V)$, π_j can be uniquely determined by setting $\pi_s = 0$ and then using (3.157)-(3.158). If π_j, thus obtained, satisfy (3.159)-(3.161), then π_j are the optimum dual variables and $(X \cup Y, L, U, V)$ is an optimum basis structure.

3.8.2 Development of the Algorithm

In this section, we present in detail the development of an algorithm for the PNCE problem. It is assumed that a value of λ, say λ_o, is known for which the PNCE problem with $y_{ij} = 0$ for each $(i,j) \varepsilon A$ has a feasible solution. Initially, the algorithm obtains an optimum basis structure for $\lambda = \lambda_o$. The value of λ is then increased and an optimum basis structure is maintained at each step. The algorithm thus produces an optimum solution for all feasible values of $\lambda \geq \lambda_o$. To obtain an optimum solution for all feasible values of $\lambda \leq \lambda_o$, r_i^* is replaced by $-r_i^*$ for each $i \varepsilon N$ and the same algorithm is applied.

A basic feasible solution for $\lambda = \lambda_o$, denoted by x_{ij}^o, can be obtained by the method described in [10]. In this basic feasible solution, let W_1, W_2 and W_3 be the sets of basic arcs and nonbasic arcs at their lower and middle bounds respectively. It can be easily verified that the basis structure $(W_1 \cup \phi, W_2, W_3, \phi)$ with the associated flow x_{ij}^o, together with the numbers $\pi_j = 0$ for each

$j \in N$, satisfies the optimality conditions and, hence, is an optimum basis structure for $\lambda = \lambda_o$.

Let $(X \cup Y, L, U, V)$ be an optimum basis structure of the PNCE problem for $\lambda = \underline{\lambda}$ with the associated flow $\underline{x}_{ij}$. If λ is increased, then the flow on arcs changes due to changing supply and demand requirements. The changed flow must satisfy the flow restrictions (3.7)-(3.11) of a feasible basis structure. Since the numbers π_j are uniquely determined for a given basis structure, the optimality conditions remain unaffected by increase in λ. Hence the highest value of λ, say $\bar{\lambda}$, upto which λ can be increased without changing the optimum basis structure is determined from the feasibility considerations (3.7)-(3.11).

It follows from Section 2.5.1 that for $\lambda \in (\underline{\lambda}, \bar{\lambda})$, flow in any arc $(i,j) \in X \cup Y$ is of the form $\underline{x}_{ij} + (\lambda - \underline{\lambda})\, z_{ij}$, whereas flow on other arcs remains unchanged. It follows from Theorem 2.6 that z_{ij} is given by

$$z_{ij} = \sum_{k \in T_j} r^*_k , \tag{3.162}$$

where T_j is the set of nodes in the resulting subtree containing node j when (i,j) is dropped from the basis. In the method to compute z_{ij} for the PNF problem described in Section 2.5.1, if we put $b^*_{kl} = 0$ for each $(k,l) \in U$, it determines z_{ij} for the PNCE problem. Having determined z_{ij}, $\bar{\lambda}$ is computed from the following inequalities:

$$0 \leq \underline{x}_{ij} + (\lambda - \underline{\lambda})\; z_{ij} \leq b_{ij}\, , \; \forall\, (i,j) \in X, \tag{3.163}$$

$$b_{ij} \leq \underline{x}_{ij} + (\lambda - \underline{\lambda})\; z_{ij} \leq a_{ij}\, , \; \forall\, (i,j) \in Y. \tag{3.164}$$

For each arc $(i,j) \in X$ define the number $\overline{\Delta\lambda}_{ij}$ as follows:

$$\overline{\Delta\lambda}_{ij} = \begin{cases} (b_{ij}-\underline{x}_{ij})/\, z_{ij} & , \text{ if } z_{ij} > 0, \\ -\underline{x}_{ij}/z_{ij} & , \text{ if } z_{ij} < 0, \\ \infty & , \text{ if } z_{ij} = 0. \end{cases} \tag{3.165}$$

Similarly, for each arc $(i,j) \in Y$ define the number $\overline{\Delta\lambda}_{ij}$ as follows:

$$\overline{\Delta\lambda}_{ij} = \begin{cases} (a_{ij}-\underline{x}_{ij})/z_{ij} & , \text{ if } z_{ij} > 0, \\ (b_{ij}-\underline{x}_{ij})/z_{ij} & , \text{ if } z_{ij} < 0, \\ \infty & , \text{ if } z_{ij} = 0. \end{cases} \tag{3.166}$$

Using (3.165) and (3.166), the conditions (3.163) and (3.164) reduce to

$$(\lambda - \underline{\lambda}) \leq \overline{\Delta\lambda}_{ij} \quad , \forall\, (i,j) \in X \cup Y. \tag{3.167}$$

Hence

$$\bar{\lambda} = \underline{\lambda} + \min_{(i,j) \in X \cup Y} \{ \overline{\Delta\lambda}_{ij} \}. \tag{3.168}$$

For all $\lambda \in (\underline{\lambda}, \bar{\lambda})$, $(X \cup Y, L, U, V)$ is the optimum basis structure and the optimum flow is given by

$$x_{ij} = \begin{cases} \underline{x}_{ij} + (\lambda - \underline{\lambda})\, z_{ij}, & \forall\, (i,j) \in X \cup Y, \\ \underline{x}_{ij} & , \forall\, (i,j) \notin X \cup Y. \end{cases} \tag{3.169}$$

Let $\bar{x}_{ij}$ be the optimum flow for $\lambda = \bar{\lambda}$. Further increase in λ is blocked by an arc (p,q) for which $\overline{\Delta\lambda}_{pq} = \min_{(i,j) \in X \cup Y} \{\overline{\Delta\lambda}_{ij}\}$.

If $(p,q) \in X$, then $\bar{x}_{pq} = 0$ or b_{pq}; and if $(p,q) \in Y$, then $\bar{x}_{pq} = b_{pq}$ or a_{pq}. If λ is increased further without changing the optimum basis structure, flow in arc (p,q) violates its bounds. Thus, a dual simplex iteration is performed to obtain an alternate optimum basis structure which may allow further increase in λ.

The dual simplex iteration is performed by dropping the arc (p,q) from the basis and selecting an arc in $\bar{L} \cup \bar{U} \cup \bar{V}$ to enter the basis, where $\bar{L}$, $\bar{U}$ and $\bar{V}$ are the sets of arcs in $L \cup U \cup V \cup \{(p,q)\}$ having $x_{ij} = 0$, b_{ij} and a_{ij} respectively. When the arc (p,q) is dropped from the basis, two subtrees are formed. Let T_p and T_q be the resulting subtrees containing node p and node q respectively. All the arcs which have their one end point in T_p and another in T_q constitute a cocycle Q. Let us define the orientation of the cocycle along (p,q) if $z_{pq} > 0$, and opposite to (p,q) if $z_{pq} < 0$. In fact, the orientation of the cocycle is defined along the direction of flow increase with λ. Let $\bar{Q}$ and $\underline{Q}$ be the sets of arcs in the cocycle Q along and opposite to its orientation respectively. (For the sake of convenience, subscripts of $\bar{Q}$ and $\underline{Q}$ are dropped).

Since the new basis must be a spanning tree, the entering arc must belong to $\bar{Q} \cup \underline{Q}$. Let $\bar{E} = (\bar{Q} \cap \bar{L}) \cup (\underline{Q} \cap \bar{U})$ and $\underline{E} = (\bar{Q} \cap \bar{U}) \cup (\underline{Q} \cap \bar{V})$. If $\bar{E} \cup \underline{E}$ is empty, then $\bar{Q} \subseteq \bar{V}$ and $\underline{Q} \subseteq \bar{L}$. In other words, $x_{ij} = a_{ij}$ for each $(i,j) \in \bar{Q}$ and $x_{ij} = 0$ for each $(i,j) \in \underline{Q}$. Clearly, the flow can not increase along the orientation of the cocycle and, hence, the PNCE problem is infeasible

for $\lambda > \bar{\lambda}$. However, if $\bar{E} \cup \underline{E}$ is nonempty, then define a number $\bar{\mu}_{kl}$ for each $(k,l) \in \bar{L} \cup \bar{U}$ and a number $\underline{\mu}_{kl}$ for each $(k,l) \in \bar{U} \cup \bar{V}$ as follows:

$$\bar{\mu}_{kl} = \begin{cases} \pi_k - \pi_l , \text{ if } (k,l) \in \bar{L} , \\ \pi_l - \pi_k , \text{ if } (k,l) \in \bar{U}, \end{cases} \tag{3.170}$$

$$\underline{\mu}_{kl} = \begin{cases} \pi_k - \pi_l + c_{kl} , \text{ if } (k,l) \in \bar{U} , \\ \pi_l - \pi_k - c_{kl} , \text{ if } (k,l) \in \bar{V}. \end{cases} \tag{3.171}$$

It follows from the optimality conditions that

$$\bar{\mu}_{kl} \geq 0, \ \forall (k,l) \in \bar{L} \cup \bar{U} \quad \text{and} \quad \underline{\mu}_{kl} \geq 0, \ \forall (k,l) \in \bar{U} \cup \bar{V}. \tag{3.172}$$

Let

$$\mu = \min. \left[\min_{(k,l) \in \bar{E}} \{\bar{\mu}_{kl}\}, \min_{(k,l) \in \underline{E}} \{\underline{\mu}_{kl}\} \right]. \tag{3.173}$$

Let this minimum be achieved for an arc (g,h) which can belong either to $\bar{E}$ or $\underline{E}$. The following theorem provides an optimum basis structure for $\lambda = \bar{\lambda}$.

<u>Theorem 3.9</u> : The basis structure obtained by replacing the arc (p,q) by the arc (g,h) as an unsaturated basic arc if $(g,h) \in \bar{E}$, or as a saturated basic arc if $(g,h) \in \underline{E}$, is an optimum basis structure for $\lambda = \bar{\lambda}$.

<u>Proof</u> : It is easy to prove using (3.157) and (3.158) that the optimum dual variables, π'_j, with respect to the new basis structure are

$$\pi'_j = \begin{cases} \pi_j \quad , \forall\, j \in T_p, \\ \pi_j + \mu, \ \forall\, j \in T_q \ , \text{ if } \ z_{pq} > 0, \\ \pi_j - \mu, \ \forall\, j \in T_q, \text{ if } \ z_{pq} < 0. \end{cases} \tag{3.174}$$

It is clear from (3.174) that if for an arc (i,j) both i and j belong either to T_p or to T_q, the optimality conditions (3.159)-(3.161) remain unaffected by changes in the optimum dual variables. Thus we need to consider the arcs belonging to $\bar{Q} \cup \underline{Q}$ only. First consider any arc $(k,l) \in \bar{Q}$. If $z_{pq} > 0$, then $\pi'_k = \pi_k$ and $\pi'_l = \pi_l + \mu$; and if $z_{pq} < 0$, then $\pi'_k = \pi_k - \mu$ and $\pi'_l = \pi_l$. In either case, substituting π'_j in (3.159)-(3.161), and then using (3.170) and (3.171), we get

$$\bar{\mu}_{kl} \geq \mu \ , \text{ if } \ (k,l) \in \bar{L} \ , \tag{3.175}$$

$$\bar{\mu}_{kl} + \mu \geq 0 \quad \text{and} \quad \underline{\mu}_{kl} \geq \mu, \text{ if } \ (k,l) \in \bar{U}, \tag{3.176}$$

$$\underline{\mu}_{kl} + \mu \geq 0 \ , \text{ if } \ (k,l) \in \bar{V}. \tag{3.177}$$

These conditions are obviously satisfied in view of (3.172) and (3.173). Similarly, it can be shown that if $(k,l) \in \underline{Q}$, the optimality conditions are satisfied.

3.8.3 Description of the Algorithm

A formal description of the algorithm for the PNCE problem is given below.

Step 1: Let $(X \cup Y, L, U, V)$ be an optimum basis structure of the PNCE problem for $\lambda = \lambda_o$. Set $\underline{\lambda} = \lambda_o$, $Z(\underline{\lambda}) = 0$ and $\pi_i = 0$, $\forall\, i \in N$. Go to Step 2.

Step 2: Let $\underline{x}_{ij}$ be the optimum flow for $\lambda = \underline{\lambda}$. Compute z_{ij}. Define the numbers $\overline{\Delta\lambda}_{ij}$ for each $(i,j) \in X \cup Y$ as follows:

$$\overline{\Delta\lambda}_{ij} = \begin{cases} (b_{ij} - \underline{x}_{ij})/z_{ij}, & \text{if } z_{ij} > 0, \\ -\underline{x}_{ij}/z_{ij}, & \text{if } z_{ij} < 0, \quad \forall (i,j) \in X, \\ \infty, & \text{if } z_{ij} = 0, \end{cases}$$

$$\overline{\Delta\lambda}_{ij} = \begin{cases} (a_{ij} - \underline{x}_{ij})/z_{ij}, & \text{if } z_{ij} > 0, \\ (b_{ij} - \underline{x}_{ij})/z_{ij}, & \text{if } z_{ij} < 0, \quad \forall (i,j) \in Y. \\ \infty, & \text{if } z_{ij} = 0, \end{cases}$$

Let $\overline{\Delta\lambda} = \min\limits_{(i,j) \in X \cup Y} \{\overline{\Delta\lambda}_{ij}\}$ and $\bar{\lambda} = \underline{\lambda} + \overline{\Delta\lambda}$. For all $\lambda \in (\underline{\lambda}, \bar{\lambda})$, the optimum solution of the PNCE problem is given by

$$x_{ij} = \begin{cases} \underline{x}_{ij} + (\lambda - \underline{\lambda})\, z_{ij}, & \forall (i,j) \in X \cup Y, \\ \underline{x}_{ij}, & \forall (i,j) \notin X \cup Y, \end{cases}$$

$$y_{ij} = \max. \{0, x_{ij} - b_{ij}\}, \quad \forall (i,j) \in A,$$

and the cost of capacity expansion is

$$Z(\lambda) = Z(\underline{\lambda}) + (\lambda - \underline{\lambda}) \sum_{(i,j) \in Y} c_{ij} z_{ij} .$$

If $\bar{\lambda} = \infty$, STOP. Otherwise update $\underline{\lambda} = \bar{\lambda}$ and identify the arc $(p,q) \in X \cup Y$ for which $\overline{\Delta\lambda}_{pq} = \overline{\Delta\lambda}$ and go to Step 3.

Step 3 : Drop the arc (p,q) from the basis. Let T_p and T_q be the resulting subtrees containing node p and node q respectively. Arcs from T_p to T_q and T_q to T_p constitute a cocycle. Define the orientation of the cocycle along (p,q) if $z_{pq} > 0$ and opposite to (p,q) if $z_{pq} < 0$. Let $\bar{Q}$ and $\underline{Q}$ be the sets of arcs in the cocycle along and opposite to its orientation respectively. Further, let $\bar{E} = (\bar{Q} \cap \bar{L}) \cup (\underline{Q} \cap \bar{U})$ and $\underline{E} = (\bar{Q} \cap \bar{U}) \cup (\underline{Q} \cap \bar{V})$, where $\bar{L}$, $\bar{U}$ and $\bar{V}$ are the sets of arcs in $L \cup U \cup V \{(p,q)\}$ having $x_{ij} = 0$, b_{ij} and a_{ij} respectively. If $\bar{E} \cup \underline{E}$ is empty, go to Step 4; otherwise define a number $\bar{\mu}_{kl}$ for each $(k,l) \in \bar{E}$ and a number $\underline{\mu}_{kl}$ for each $(k,l) \in \underline{E}$ as follows:

$$\bar{\mu}_{kl} = \begin{cases} \pi_k - \pi_l, & \text{if } (k,l) \in \bar{L}, \\ \pi_l - \pi_k, & \text{if } (k,l) \in \bar{U}, \end{cases}$$

$$\underline{\mu}_{kl} = \begin{cases} \pi_k - \pi_l + c_{kl}, & \text{if } (k,l) \in \bar{U}, \\ \pi_l - \pi_k - c_{kl}, & \text{if } (k,l) \in \bar{V}. \end{cases}$$

Let $\mu = \min. \left[\min_{(k,l) \in \bar{E}} \{ \bar{\mu}_{kl} \}, \min_{(k,l) \in \underline{E}} \{ \underline{\mu}_{kl} \} \right]$. Let (g,h) be the arc which achieves this minimum. If $(g,h) \in \bar{E}$, then (g,h) enters the basis as an unsaturated basic arc; otherwise it enters the basis as a saturated basic arc. Update the basis structure. Further, update the dual variables as

$$\pi_j = \begin{cases} \pi_j, & \forall\, j \in T_p, \\ \pi_j + \mu, & \forall\, j \in T_q, \text{ if } z_{pq} > 0, \\ \pi_j - \mu, & \forall\, j \in T_q, \text{ if } z_{pq} < 0, \end{cases}$$

Step 4 : The PNCE problem is infeasible for $\lambda > \bar{\lambda}$. STOP.

3.8.4 Numerical Example

We now solve a numerical example to illustrate the various steps of the PNCE algorithm. The network is shown in Fig. 3.4. The numbers r_i^o and r_i^* are indicated above the node i. The numbers a_{ij}, b_{ij} and c_{ij} are mentioned over the arc (i,j). The problem is to be solved for all feasible values of $\lambda \geq 0$.

Solution of the first six iterations is given in Table 3.3. In the table, the symbol ↑ denotes the arc entering the basis and ↓ denotes the basic arc leaving the basis. Basis in various iterations are shown in Fig. 3.5. In this figure, saturated basic arcs are drawn in double lines.

3.8.5 Capacity Expansion to Maximize Flow

The problem of allocating a given budget D to increase capacities of various arcs so that the flow from source to sink is maximized, can be formulated as follows:

$$\text{Maximize} \quad v, \tag{3.178}$$

subject to

$$\sum_{(j,i)\in I(i)} x_{ji} - \sum_{(i,j)\in O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i\neq s,t, \ \forall\, i \in N, \\ v, & \text{if } i=t, \end{cases} \tag{3.179}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij}, \ \forall\, (i,j) \in A, \tag{3.180}$$

$$\sum_{(i,j)\in A} c_{ij}\, y_{ij} \leq D, \tag{3.181}$$

where $c_{ij} > 0$ for each $(i,j) \in A$.

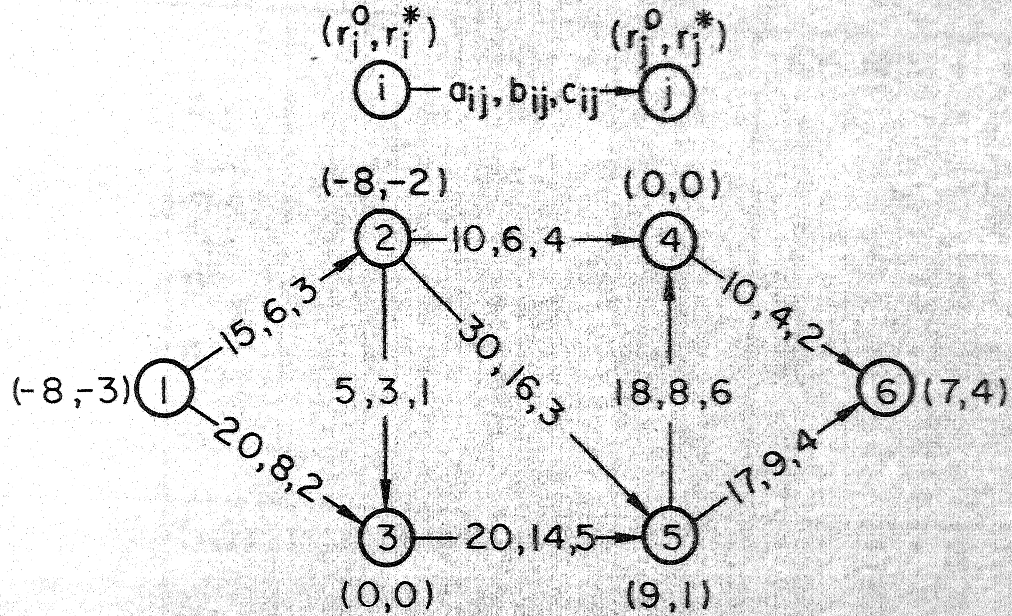

Fig. 3.4 Numerical example for the PNCE problem

Table 3.3: Solution of the network flow problem

Iteration No.	$\underline{\lambda}$	$Z(\underline{\lambda})$		Basic arcs					Nonbasic arcs			
			(i,j)	(1,2)	(3,5)	(2,5)	(4,6)	(5,6)	(1,3)	(2,3)	(2,4)	(5,4)
			$\underline{x}_{ij}$	0	8	8	0	7	8	0	0	0
1.	0	0	Status	X	X	X	X	X	U	L	L	L
			z_{ij}	3	0	5	0	4	–	–	–	–
			$\overline{\Delta\lambda}_{ij}$	2	∞	1.6	∞	0.5	–	–	–	–
			$\bar{\mu}_{ij}$	–	–	–	–	–	–	–	0	0
			$\underline{\mu}_{ij}$	–	–	–	–	4 ↓	–	–	– ↑	–
			(i,j)	(1,2)	(3,5)	(2,5)	(4,6)	(2,4)	(1,3)	(2,3)	(5,6)	(5,4)
			$\underline{x}_{ij}$	1.5	8	10.5	0	0	8	0	9	0
			Status	X	X	X	X	X	U	L	U	L
2.	0.5	0	z_{ij}	3	0	1	4	4	–	–	–	–
			$\overline{\Delta\lambda}_{ij}$	1.5	∞	5.5	1	1.5	–	–	–	–
			$\bar{\mu}_{ij}$	–	–	–	–	–	–	–	–	–
			$\underline{\mu}_{ij}$	–	–	–	2 ↓↑	–	–	–	4	–
			(i,j)	(1,2)	(3,5)	(2,5)	(4,6)	(2,4)	(1,3)	(2,3)	(5,6)	(5,4)
			$\underline{x}_{ij}$	4.5	8	11.5	4	4	8	0	9	0
			Status	X	X	X	Y	X	U	L	U	L
3.	1.5	0	z_{ij}	3	0	1	4	4	–	–	–	–
			$\overline{\Delta\lambda}_{ij}$	0.5	∞	4.5	1.5	0.5	–	–	–	–
			$\bar{\mu}_{ij}$	–	–	–	–	–	–	–	–	0
			$\underline{\mu}_{ij}$	–	–	–	–	4 ↑	–	–	2	– ↓

Iteration No.	$\underline{\lambda}$	$Z(\underline{\lambda})$		Basic arcs					Nonbasic arcs			
			(i,j)	(1,2)	(3,5)	(2,5)	(4,6)	(5,4)	(1,3)	(2,3)	(5,6)	(2,4)
			$\underline{x}_{ij}$	6	8	12	6	0	8	0	9	6
4.	2	4	Status	X	X	X	Y	X	U	L	U	U
			z_{ij}	3	0	5	4	4	–	–	–	–
			$\overline{\Delta\lambda}_{ij}$	0	∞	0.8	1	2	–	–	–	–
			$\bar{\mu}_{ij}$	–	–	–	–	–	–	–	–	–
			$\underline{\mu}_{ij}$	3 ↓	–	–	–	–	2 ↑	–	–	–
			(i,j)	(1,3)	(3,5)	(2,5)	(4,6)	(5,4)	(1,2)	(2,3)	(5,6)	(2,4)
			$\underline{x}_{ij}$	8	8	12	6	0	6	0	9	6
5.	2	4	Status	Y	X	X	Y	X	U	L	U	U
			z_{ij}	3	3	2	4	4	–	–	–	–
			$\overline{\Delta\lambda}_{ij}$	4	2	2	1	2	–	–	–	–
			$\bar{\mu}_{ij}$	–	–	–	–	–	–	–	–	–
			$\underline{\mu}_{ij}$	–	–	–	– ↓	–	–	–	2 ↑	–
			(i,j)	(1,3)	(3,5)	(2,5)	(5,6)	(5,4)	(1,2)	(2,3)	(4,6)	(2,4)
			$\underline{x}_{ij}$	11	11	14	9	0	6	0	10	6
6.	3	18	Status	Y	X	X	Y	X	U	L	V	U
			z_{ij}	3	3	2	4	0	–	–	–	–
			$\overline{\Delta\lambda}_{ij}$	3	1	1	2	∞	–	–	–	–
			$\bar{\mu}_{ij}$	–	–	–	–	–	–	–	–	–
			$\underline{\mu}_{ij}$	–	5 ↓	–	–	–	1 ↑	–	–	–

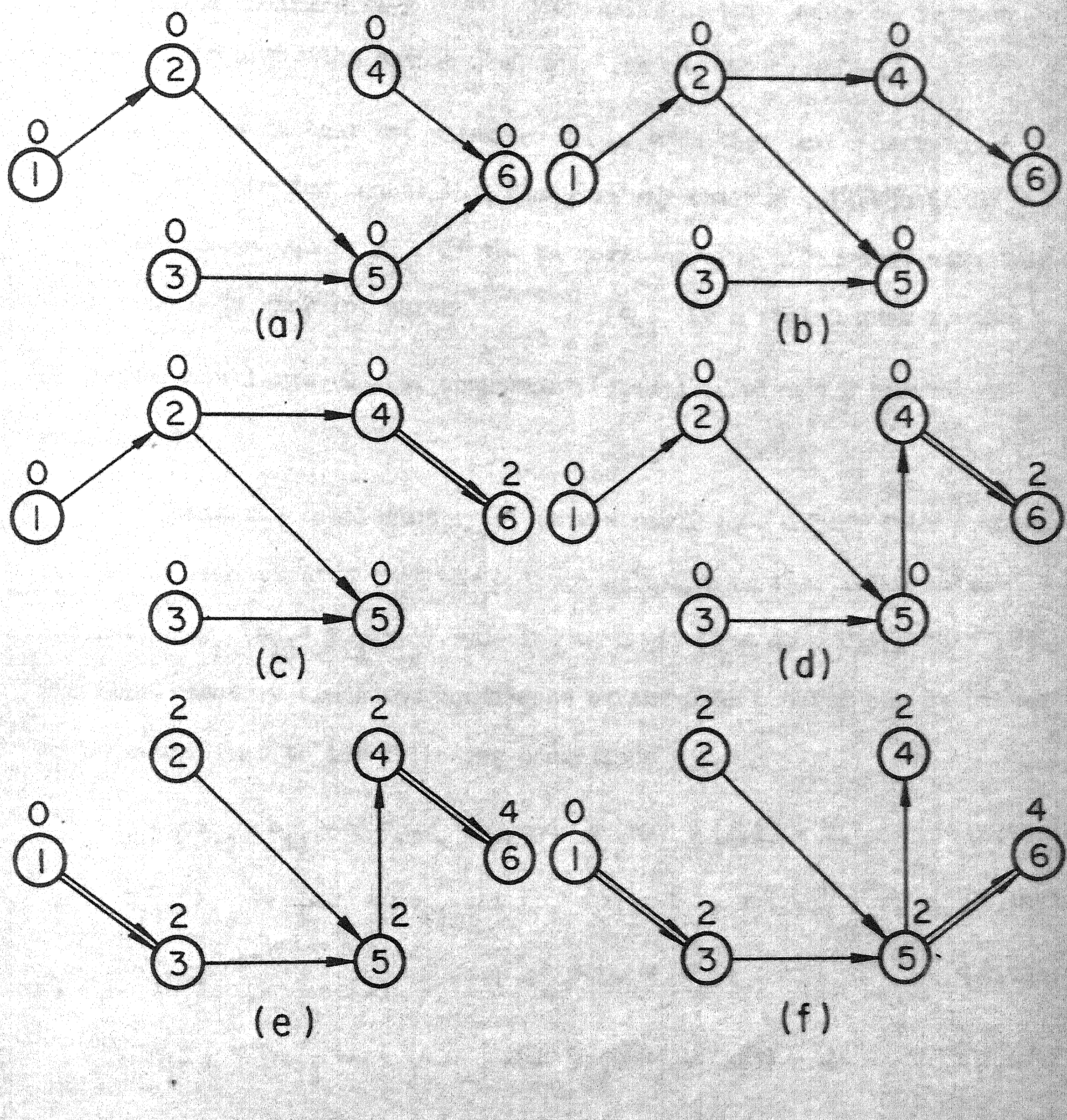

Fig. 3.5 Basis in various iterations

We refer to the problem (3.178)-(3.181) as the <u>Capacity Expansion to Maximize Flow</u> (CEMF) problem. In this section, we show how the PNCE algorithm can be used to solve the CEMF problem.

We assume that the value of D is such that the constraint (3.181) is a binding constraint, i.e., in any optimum solution it is satisfied as an equality. If the network contains a directed path P from source to sink for which $\sum_{(i,j) \varepsilon P} c_{ij}$ is a finite number, then no matter how large D is, constraint (3.181) is always a binding constraint.

Using the complementary slackness conditions of the CEMF problem, it can be demonstrated that the optimum solution of the CEMF problem satisfies $y_{ij}(b_{ij}+y_{ij}-x_{ij}) = 0$ for each $(i,j) \varepsilon A$. Using this result, the complementary slackness conditions of the CEMF problem can be proved to be equivalent to the following conditions :

$$0 < x_{ij} < b_{ij} \Rightarrow \sigma_i - \sigma_j = 0 \text{ and } y_{ij} = 0, \forall (i,j) \varepsilon A, \tag{3.182}$$

$$b_{ij} < x_{ij} \Rightarrow \sigma_i - \sigma_j = \mu c_{ij} \text{ and } y_{ij} = x_{ij} - b_{ij}, \forall (i,j) \varepsilon A, \tag{3.183}$$

$$\sigma_i - \sigma_j < 0 \Rightarrow x_{ij} = 0 \text{ and } y_{ij} = 0, \forall (i,j) \varepsilon A, \tag{3.184}$$

$$0 < \sigma_i - \sigma_j < \mu c_{ij} \Rightarrow x_{ij} = b_{ij} \text{ and } y_{ij} = 0, \forall (i,j) \varepsilon A, \tag{3.185}$$

$$v > 0 \Rightarrow \sigma_s - \sigma_t = 1, \tag{3.186}$$

$$\mu(D - \sum_{(i,j) \varepsilon A} c_{ij} y_{ij}) = 0, \tag{3.187}$$

where σ_i and $\mu > 0$ are the dual variables associated with the constraints (3.179) and (3.180) respectively.

Substituting $\pi_i = -\sigma_i/\mu$ for each $i \in N$ in (3.182)-(3.187), we obtain the following equivalent conditions:

$$0 < x_{ij} < b_{ij} \Longrightarrow \pi_j - \pi_i = 0 \text{ and } y_{ij} = 0 \text{ , } \forall (i,j) \in A, \tag{3.188}$$

$$b_{ij} < x_{ij} \Longrightarrow \pi_j - \pi_i = c_{ij} \text{ and } y_{ij} = x_{ij} - b_{ij}, \ \forall (i,j) \in A, \tag{3.189}$$

$$\pi_j - \pi_i < 0 \Rightarrow x_{ij} = 0 \text{ and } y_{ij} = 0, \ \forall (i,j) \in A, \tag{3.190}$$

$$0 < \pi_j - \pi_i < c_{ij} \Rightarrow x_{ij} = b_{ij} \text{ and } y_{ij} = 0, \ \forall (i,j) \in A, \tag{3.191}$$

$$v > 0 \Longrightarrow \pi_t - \pi_s = 1/\mu \text{ ,} \tag{3.192}$$

$$\mu \, (D - \sum_{(i,j) \in A} c_{ij} \, y_{ij}) = 0. \tag{3.193}$$

We note that the conditions (3.188)-(3.191) are included in the complementary slackness conditions of the PNCE problem. The condition (3.192) can always be satisfied by setting $\pi_s = 0$ and $\mu = 1/\pi_t$. Thus, (3.193) is the only additional condition.

In the PNCE problem, let us set

$$r_i^0 = 0 \text{ , } \forall i \in N, \tag{3.194}$$

$$r_i^* = \begin{cases} -1 \text{ , } & \text{if } i = s, \\ 0 \text{ , } & \text{if } i \neq s,t, \ \forall i \in N, \\ 1 \text{ , } & \text{if } i = t, \end{cases} \tag{3.195}$$

$$a_{ij} = \infty, \ \forall (i,j) \in A, \tag{3.196}$$

and consider the optimum solution of the PNCE problem when $Z(\lambda) = D$. This solution satisfies (3.193) and hence is an optimum solution of the CEMF problem.

When the PNCE algorithm is applied to the CEMF problem, then at each iteration it augments the flow from source to sink on the path consisting of basic arcs and capacities of all saturated basic arcs lying on this path are increased by the amount of augmented flow. The algorithm terminates when either the given budget is exhausted or a path consisting solely of saturated basic arcs is discovered implying that infinite amount of flow can be augmented on this path.

Fulkerson [39] solves the CEMF problem by solving a sequence of minimum cost flow problems. At each iteration, a labelling routine is executed to increase the flow from source to sink and when the flow can not increase, the value of π_t is incremented by one unit. In our algorithm, flow is augmented on the known path from source to sink and in two subsequent iterations, the value of π_t can increase by more than one unit. Hence our algorithm requires less iterations and in each iteration less computations are performed.

Hu [52] has also proposed a method to solve the CEMF problem. At each iteration, his method defines arc lengths and augments the flow on the shortest path from source to sink. His method is similar to our algorithm. The essential difference between the two is that shortest paths are implicitly enumerated by our algorithm, whereas these are obtained by solving shortest path problems in Hu's method. Hence our algorithm is computationally superior to Hu's method.

CHAPTER IV

CONVEX COST NETWORK FLOW AND RELATED PROBLEMS

4.1 *INTRODUCTION*

In this chapter, we study the minimum cost flow problem when cost of flow over each arc is given by a piecewise linear convex function. We refer to this problem as the Convex Cost Network Flow (CCNF) problem.

Several practical problems can be formulated as the CCNF problem. Cost structure of water transportation in pipeline networks is convex [33]. Transportation problem, with uncertain demands and a penalty for short supply, can be formulated as the CCNF problem [21, 34]. Capacity expansion of networks [64], and a problem associated with transmission of power in electrical networks [51] are also CCNF problems.

The CCNF problem has been studied by several researchers. The following three approaches have been reported in the literature to handle such problems:

(i) Primal-Dual Approach : This approach assumes cost functions to be piecewise linear convex functions and is essentially a generalization of the out-of-kilter method. An algorithm based on this approach is suggested by Minty [70,71]. Lawler [64] also discusses this approach.

(ii) Shortest Path Approach : This approach consists of determining, at each iteration, a shortest path from source to sink based on incremental costs of arcs and augmenting flow over it. Hu's algorithm [51] is based

on this approach. The algorithm assumes that all cost functions are linearized along segments of any common length d, such that all capacities are multiples of d.

(iii) Negative Cycle Approach : In this approach, the objective function is improved in successive feasible solutions by augmenting flows in negative cycles based on incremental costs. When no such improvement is possible, the solution is optimum. Several authors have used this approach and the essential difference is in the manner in which negative cycles are determined. For piecewise linear convex functions, Florian and Roberts [36] and Fillet et. al [35] have offered node labelling methods, and Beale [11] and Klein [59] have suggested matrix manipulation methods to identify negative cycles. For continuous convex functions, enumeration is suggested by Menon [67]. A more efficient method is proposed by Weintraub [93] which identifies negative cycles by solving a finite number of assignment problems.

It is wellknown that the CCNF problem can be transformed to the minimum cost flow problem by introducing one variable for each linear segment. This transformation enlarges the network substantially. We note that in the optimum solution all the additional variables can be considered implicitly. This observation allows us to define the concept of optimum basis structure for the CCNF problem. The optimum basis structure is then used to parametrize v, the flow to be transshipped from source to sink. Initially, the algorithm obtains an optimum basis structure for $v=0$. The value of v is then increased and an optimum

basis structure is maintained at each step until either the desired flow is established or infeasibility of the CCNF problem is discovered. Intuitively speaking, the algorithm implicitly emmerates shortest paths from source to sink and augments flow over these paths. Hence the algorithm belongs to the shortest path approach. The computational complexity of the algorithm is shown to be O(mnv).

The time-cost tradeoff problem in a CPM network is of considerable practical significance. On account of its importance, the problem has been studied by several researchers [12, 20, 31, 40, 46, 58, 63, 78, 80, 84]. We show that the CCNF algorithm can be used to solve this problem.

We also establish a relationship between the CCNF problem and the maximum flow problem with piecewise linear concave gain functions. This relationship is a generalization of the relationship between the minimum cost flow problem and the maximum flow problem with gains recognized by several researchers [66, 74, 90]. We suggest modifications in the CCNF algorithm in order to solve this problem.

The problem of optimally allocating a given budget to increase the capacities of various arcs to maximize the flow in a network is considered by several researchers for different capacity expansion costs [9, 39, 50, 52, 79]. We show that when capacity expansion costs are given by piecewise linear convex functions, this problem can be solved by the CCNF algorithm.

A computer program was written for the CCNF algorithm. The computational performance of the program is reported. The program solves CCNF problems with 200 nodes, 2000 arcs and cost functions consisting of 5 linear segments in 37 seconds.

4.2 PROBLEM STATEMENT

The mathematical statement of the CCNF problem is as follows:

$$\text{Minimize } Z = \sum_{(i,j) \in A} C_{ij}(x_{ij}) \,, \tag{4.1}$$

subject to

$$\sum_{(j,i) \in I(i)} x_{ji} - \sum_{(i,j) \in O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i \neq s,t, \ \forall\, i \in N, \\ v, & \text{if } i=t, \end{cases} \tag{4.2}$$

$$x_{ij} \geq 0, \ \forall\, (i,j) \in A, \tag{4.3}$$

where $C_{ij}(x_{ij})$ is a piecewise linear convex function for each $(i,j) \in A$.

It may be noted that capacity restrictions are not explicitly stated because they can be incorporated in $C_{ij}(x_{ij})$.

Let $C_{ij}(x_{ij})$ consist of $p(i,j)$ linear segments. Let $p = \max_{(i,j) \in A} \{p(i,j)\}$. Associated with each arc $(i,j) \in A$ are numbers $0 = b_{ij}^{0} < b_{ij}^{1} < \ldots < b_{ij}^{p(i,j)} = b_{ij}^{p(i,j)+1} = \ldots = b_{ij}^{p}$ and slope of $C_{ij}(x_{ij})$ in the interval $(b_{ij}^{k-1}, b_{ij}^{k})$ is c_{ij}^{k}. We assume that $c_{ij}^{1} < c_{ij}^{2} < \ldots < c_{ij}^{p}$ which implies that $C_{ij}(x_{ij})$ is a convex function. Further, it is assumed that $C_{ij}(0) = 0$ for each $(i,j) \in A$.

4.3 OPTIMALITY CONDITIONS

Let us partition A into the sets $B^1, B^2, \ldots, B^p$, $U^o, U^1, \ldots, U^p$ such that $B^1 \cup B^2 \cup \ldots \cup B^p$ is a spanning tree. If we set

$$x_{ij} = b_{ij}^k \quad , \quad \forall (i,j) \in U^k, \; \forall k = 0,1,\ldots,p, \tag{4.4}$$

then there exists a unique flow x_{ij} which satisfies the flow conservation equations (4.2). If this flow further satisfies

$$b_{ij}^{k-1} \leq x_{ij} \leq b_{ij}^k \; , \; \forall (i,j) \in B^k, \quad \forall k=1,\ldots,p, \tag{4.5}$$

then we refer to $B^1 \cup B^2 \cup \ldots \cup B^p$ as a <u>feasible basis</u> and $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$ as a <u>feasible basis structure</u> of the CCNF problem. We refer to an arc $(i,j) \in A$ as a <u>k-basic arc</u> if $(i,j) \in B^k$ and a <u>k-nonbasic arc</u> if $(i,j) \in U^k$. We define an <u>optimum basis structure</u> as a feasible basis structure for which the associated flow x_{ij} is the optimum solution of the CCNF problem. In this section, we derive the necessary and sufficient conditions for a feasible basis structure to be an optimum basis structure.

It is wellknown [72] that the CCNF problem can be transformed to a linear program by introducing the variables $y_{ij}^1, y_{ij}^2, \ldots, y_{ij}^p$ for each $(i,j) \in A$ and replacing each x_{ij} by $\sum_{k=1}^{p} y_{ij}^k$. The CCNF problem then becomes

$$\text{Minimize } Z = \sum_{(i,j) \in A} \; \sum_{k=1}^{p} c_{ij}^k \, y_{ij}^k \; , \tag{4.6}$$

subject to

$$\sum_{(j,i) \in I(i)} \; \sum_{k=1}^{p} y_{ji}^k - \sum_{(i,j) \in O(i)} \; \sum_{k=1}^{p} y_{ij}^k = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i \neq s,t, \; \forall i \in N, \\ v, & \text{if } i=t, \end{cases} \tag{4.7}$$

$$0 \le y_{ij}^{k} \le b_{ij}^{k} - b_{ij}^{k-1}, \ \forall (i,j) \in A, \ \forall k = 1,\ldots,p. \qquad (4.8)$$

It is also known [72] that any optimum solution of (4.6)-(4.8) satisfies the following conditions:

$$y_{ij}^{k} > 0 \Longrightarrow y_{ij}^{\ell} = b_{ij}^{\ell} - b_{ij}^{\ell-1}, \ \forall \ \ell = 1,\ldots,k-1, \qquad (4.9)$$

$$y_{ij}^{k} < b_{ij}^{k} - b_{ij}^{k-1} \Longrightarrow y_{ij}^{\ell} = 0 \quad , \ \forall \ \ell = k+1,\ldots,p. \qquad (4.10)$$

The dual of (4.6)-(4.8) is

$$\text{Maximize} \quad v(\pi_t - \pi_s) - \sum_{(i,j) \in A} \sum_{k=1}^{p} (b_{ij}^{k} - b_{ij}^{k-1}) \, \delta_{ij}^{k}, \qquad (4.11)$$

subject to

$$\pi_j - \pi_i - \delta_{ij}^{k} \le c_{ij}^{k}, \ \forall (i,j) \in A, \ \forall \ k=1,\ldots,p, \qquad (4.12)$$

$$\delta_{ij}^{k} \ge 0, \ \forall (i,j) \in A, \ \forall k=1,\ldots,p, \qquad (4.13)$$

and the complementary slackness conditions are

$$0 < y_{ij}^{k} < b_{ij}^{k} - b_{ij}^{k-1} \Longrightarrow \pi_j - \pi_i = c_{ij}^{k}, \ \forall (i,j) \in A, \ \forall k=1,\ldots,p, \qquad (4.14)$$

$$\pi_j - \pi_i < c_{ij}^{k} \quad \Longrightarrow \quad y_{ij}^{k} = 0, \ \forall (i,j) \in A \quad , \ \forall k=1,\ldots,p, \qquad (4.15)$$

$$\pi_j - \pi_i > c_{ij}^{k} \quad \Longrightarrow \quad y_{ij}^{k} = b_{ij}^{k} - b_{ij}^{k-1}, \ \forall (i,j) \in A, \ \forall k=1,\ldots,p. \qquad (4.16)$$

These conditions, in view of (4.9) and (4.10), can be written as

$$0 < y_{ij}^{k} < b_{ij}^{k} - b_{ij}^{k-1} \Longrightarrow \pi_j - \pi_i = c_{ij}^{k}, \ \forall (i,j) \in A, \ \forall k=1,\ldots,p, \qquad (4.17)$$

$$c_{ij}^{k} < \pi_j - \pi_i < c_{ij}^{k+1} \Longrightarrow y_{ij}^{\ell} = b_{ij}^{\ell} - b_{ij}^{\ell-1}, \ \forall \ \ell = 1,2,\ldots,k, \ \text{and}$$

$$y_{ij}^{\ell} = 0, \ \forall \ \ell = k+1,\ldots,p, \ \forall (i,j) \in A, \ k=0,\ldots,p. \qquad (4.18)$$

where $c_{ij}^{0} = -\infty$ and $c_{ij}^{p+1} = \infty$ for each $(i,j) \in A$.

The conditions (4.17) and (4.18), in terms of x_{ij}, are

$$b_{ij}^{k-1} < x_{ij}^{k} < b_{ij}^{k} \Longrightarrow \pi_j - \pi_i = c_{ij}^{k}, \ \forall\,(i,j) \in A, \ \forall\, k=1,\ldots,p, \tag{4.19}$$

$$c_{ij}^{k} < \pi_j - \pi_i < c_{ij}^{k+1} \Longrightarrow x_{ij} = b_{ij}^{k}, \ \forall\,(i,j) \in A, \ \forall\, k=0,\ldots,p. \tag{4.20}$$

It follows from (4.19) and (4.20) that the necessary and sufficient conditions for a feasible basis structure $(B^1 \cup B^2 \cup \ldots \cup B^p,\ U^o, U^1, \ldots, U^p)$ to be an optimum basis structure are that there exists numbers π_j satisfying the following conditions:

$$\pi_j - \pi_i = c_{ij}^{k}, \ \forall\,(i,j) \in B^k, \ \forall\, k=1,\ldots,p, \tag{4.21}$$

$$c_{ij}^{k} \le \pi_j - \pi_i \le c_{ij}^{k+1}, \ \forall\,(i,j) \in U^k, \ \forall\, k=0,1,\ldots,p. \tag{4.22}$$

We refer to these conditions as the optimality conditions for the CCNF problem. We refer to π_j satisfying the optimality conditions as the optimum dual variables associated with the optimum basis structure $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$.

It may be noted that given any feasible basis structure $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$, π_j can be uniquely computed by setting $\pi_s = 0$ and then using (4.21). If π_j thus obtained satisfy (4.22), then π_j are the optimum dual variables and $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$ is the optimum basis structure.

4.4 *DEVELOPMENT OF THE ALGORITHM*

The algorithm for the CCNF problem treats v as a parameter and maintains an optimum basis structure at every step. Initially, an

optimum basis structure for $v=0$ is obtained. The algorithm then determines v_1 such that this basis structure remains optimum for all $v \in (0,v_1)$. A dual simplex iteration is then performed to obtain an alternate optimum basis structure for $v=v_1$, which may allow further increase in v. The algorithm proceeds in this manner until either the desired flow is established or infeasibility of the problem is indicated.

4.4.1 Initial Optimum Basis Structure

We now make one assumption which, though not very restrictive, simplifies the computations considerably.

Assumption 4.1 : The network does not contain any negative cycle with c^1_{ij} as the length of each arc $(i,j) \in A$.

This assumption implies that $x_{ij} = 0$ for each $(i,j) \in A$, is the optimum flow for $v=0$. Let T be the tree of shortest paths rooted at source with c^1_{ij} as the length of each arc $(i,j) \in A$. Further, let π_j be the length of the shortest path from source to any node $j \in N$. Then we know [22] that

$$\pi_j - \pi_i = c^1_{ij} \quad , \forall \ (i,j) \in T, \tag{4.23}$$

$$\pi_j - \pi_i \leq c^1_{ij} \quad , \forall \ (i,j) \in \hat{T}, \tag{4.24}$$

where $\hat{T} = A - T$.

In view of (4.23) and (4.24), it follows that $(B^1 \cup B^2 \cup \dots \cup B^p, U^0, U^1, \dots, U^p)$ is an optimum basis structure for $v=0$, where $B^1 = T$, $U^0 = \hat{T}$ and $B^2 = \dots = B^p = U^1 = \dots = U^p = \phi$.

4.4.2 Characteristic Interval

Let $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$ be an optimum basis structure for $v = \underline{v}$ with the associated flow $\underline{x}_{ij}$. Let $\underline{Z}$ be the cost of this flow. Further, let π_j be the optimum dual variables associated with this basis structure. We now determine the maximum value of v, say $\bar{v}$, such that the current basis structure remains optimum for any $v \in (\underline{v}, \bar{v})$. The interval $(\underline{v}, \bar{v})$ is known as the characteristic interval associated with $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$.

In the basis, there is a unique path from source to sink. We refer to this path as the basic path. Let P be the basic path in B, and $\bar{P}$ and $\underline{P}$ be the sets of forward and backward arcs in P respectively. The value of v can be increased by sending additional flow from source to sink through the basic path. We assume, without any loss of generality, that $\pi_s = 0$. Then, using (4.21) it can be shown that

$$\pi_t = \sum_{k=1}^{p} \sum_{(i,j) \in \bar{P} \cap B^k} c_{ij}^k - \sum_{k=1}^{p} \sum_{(i,j) \in \underline{P} \cap B^k} c_{ij}^k . \qquad (4.25)$$

Thus, π_t denotes the cost of sending unit additional flow from source to sink through the basic path.

Since π_j are uniquely determined for a given basis structure, the optimality conditions remain unaffected by increase in v. However, flow on arcs belonging to P changes and the changed flow must satisfy the bound restrictions (4.5) of the basic arcs. We now determine the maximum additional flow, denoted by $\bar{w}$, that can be sent without violating the bound restrictions. If we define a number $\bar{w}_{ij}$ for each $(i,j) \in P$ as

$$\bar{w}_{ij} = \begin{cases} b^k_{ij} - \underline{x}_{ij}\ , \text{ if } (i,j) \in \bar{P} \cap B^k, \\ \underline{x}_{ij} - b^{k-1}_{ij}, \text{ if } (i,j) \in \underline{P} \cap B^k, \end{cases} \tag{4.26}$$

then

$$\bar{w} = \min_{(i,j) \in P} \{\bar{w}_{ij}\}\ . \tag{4.27}$$

Hence

$$\bar{v} = \underline{v} + \bar{w}\ . \tag{4.28}$$

For all $v \in (\underline{v}, \bar{v})$, $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$ is the optimum basis structure and the optimum flow is given by

$$x_{ij} = \begin{cases} \underline{x}_{ij} + \theta\,\bar{w}\ , \ \forall\,(i,j) \in \bar{P}\ , \\ \underline{x}_{ij} - \theta\,\bar{w}\ , \ \forall\,(i,j) \in \underline{P}\ , \\ \underline{x}_{ij}\ , \ \forall\,(i,j) \notin P\ , \end{cases} \tag{4.29}$$

and the cost of this flow is

$$Z = \underline{Z} + \theta\,\bar{w}\,\pi_t \tag{4.30}$$

where $\theta = (v - \underline{v}) / (\bar{v} - \underline{v})$.

Let $\bar{x}_{ij}$ denote the optimum flow for $v = \bar{v}$. Further increase in v is blocked by an arc $(\alpha,\beta) \in P$ for which $\bar{w}_{\alpha\beta} = \bar{w}$. If there are more than one arcs satisfying this criteria, then let (α,β) be the arc which is nearest to the source. Let $(\alpha,\beta) \in B^e$. If $(\alpha,\beta) \in \bar{P}$, then $\bar{x}_{\alpha\beta} = b^e_{\alpha\beta}$; and if $(\alpha,\beta) \in \underline{P}$, then $\bar{x}_{\alpha\beta} = b^{e-1}_{\alpha\beta}$. If v is increased further without changing the basis structure, flow in arc (α,β) violates its respective bound. Thus, a dual simplex iteration is performed to obtain an alternate optimum basis structure for $v = \bar{v}$, which may allow further increase in v.

4.4.3 Dual Simplex Iteration

The dual simplex iteration is performed by dropping the arc (α,β) from the basis and selecting an arc belonging to $U^o \cup U^1 \cup \ldots \cup U^p \{(\alpha,\beta)\}$ to enter the basis. It may be noted that (α,β) can also enter the basis, but at a different status, i.e., it would either be a $(e+1)$-basic arc or a $(e-1)$-basic arc. When (α,β) is dropped from the basis, two subtrees are formed. Let T_s and T_t be the resulting subtrees containing source and sink respectively. Arcs from T_s to T_t and T_t to T_s constitute a cutset Q. Let $\bar{Q}$ and $\underline{Q}$ be the sets of forward and backward arcs in Q respectively. Since the new basis must be a spanning tree, the entering arc must belong to Q. Let us define the following sets for convenience:

$$\hat{U}^k = \{(i,j) \in U^o \cup U^1 \cup \ldots \cup U^p \cup \{(\alpha,\beta)\}: x_{ij} = b^k_{ij}\}, \; \forall\, k=0,\ldots,p, \tag{4.31}$$

$$\hat{B}^k = \begin{cases} B^k - \{(\alpha,\beta)\}, & \text{if } k=e, \\ B^k, & \text{if } k\neq e \end{cases}, \; \forall\, k=1,\ldots,p. \tag{4.32}$$

We follow the convention that if $x_{ij} = b^p_{ij}$, then $(i,j) \in \hat{U}^p$. Let us define the numbers $\bar{\mu}_{h\ell}$ and $\underline{\mu}_{h\ell}$ for each $(h,\ell) \in \hat{U}^k$, $\forall\, k=0,1,\ldots,p$ as follows:

$$\bar{\mu}_{h\ell} = \pi_h - \pi_\ell + c^{k+1}_{h\ell}, \tag{4.33}$$

$$\underline{\mu}_{h\ell} = \pi_\ell - \pi_h - c^k_{h\ell}. \tag{4.34}$$

It follows from the optimality conditions that

$$\bar{\mu}_{h\ell} \geq 0 \quad \text{and} \quad \underline{\mu}_{h\ell} \geq 0, \; \forall\, (h,\ell) \;\varepsilon\; Q. \tag{4.35}$$

Let

$$\mu \;= \min.\; [\min_{(h,\ell)\,\varepsilon\,\bar{Q}} \{\bar{\mu}_{h\ell}\},\; \min_{(h,\ell)\,\varepsilon\,\underline{Q}} \{\underline{\mu}_{h\ell}\}]. \tag{4.36}$$

If $\mu = \infty$, then $\bar{Q} \subseteq \hat{U}^p$ and $\underline{Q} \subseteq \hat{U}^o$. This implies that all the forward arcs in Q have $x_{ij} = b^p_{ij}$ and all the backward arcs in Q have $x_{ij} = 0$. Clearly, flow in the network equals the maximum flow with b^p_{ij} as the capacity of each arc $(i,j) \;\varepsilon\; A$ and the CCNF problem is infeasible for $v > \bar{v}$. However, if $\mu \neq \infty$, then let the minimum in (4.36) be achieved at an arc (q,r) which can belong either to $\bar{Q}$ or $\underline{Q}$. Let $(q,r) \;\varepsilon\; \hat{U}^f$. The following theorem provides an alternate optimum basis structure for $v = \bar{v}$.

<u>Theorem 4.1</u> : The basis structure obtained by replacing the arc (α,β) by (q,r) as a $(f+1)$-basic arc if $(q,r) \;\varepsilon\; \bar{Q}$, or as a f-basic arc if $(q,r) \;\varepsilon\; \underline{Q}$ is an optimum basis structure for $v = \bar{v}$.

<u>Proof</u> : It can be easily shown using the optimality condition (4.21) that the dual variables, π'_j, with respect to the new basis structure are

$$\pi'_j = \begin{cases} \pi_j \;, & \forall\, j \;\varepsilon\; T_s, \\ \pi_j + \mu, & \forall\, j \;\varepsilon\; T_t. \end{cases} \tag{4.37}$$

For any arc $(i,j) \;\varepsilon\; A$, if both i and j belong either to T_s or to T_t, the optimality condition (4.22) is obviously satisfied. Thus, we need to consider arcs belonging to Q only. Consider any arc $(h,\ell) \;\varepsilon\; \bar{Q}$. Clearly, $\pi'_h = \pi_h$ and $\pi'_\ell = \pi_\ell + \mu$. Substituting these values in (4.22) and then using (4.33) and (4.34), we get

$$\bar{\mu}_{h\ell} \geq \mu , \tag{4.38}$$

$$\underline{\mu}_{h\ell} + \mu \geq 0, \tag{4.39}$$

which are clearly satisfied in view of (4.35) and (4.36). Similarly, it can be shown that if $(h, \ell) \in \underline{Q}$, the optimality conditions are satisfied.

4.4.4 Physical Interpretation

Initially, the CCNF algorithm augments the flow on the shortest path from source to sink. The flow blocking arc is dropped from the basis and a cutset results. An arc which belongs to this cutset and yields minimum increase in π_t is selected to enter the basis and thereby giving a new path from source to sink on which flow can be augmented. Since π_t is the cost of unit augmentation from source to sink, the new path is essentially a shortest path with respect to the changed flow. Therefore, the algorithm implicitly enumerates shortest paths from source to sink and augments flow over these paths.

Hu's algorithm [51] for the CCNF problem defines appropriate arc lengths at each iteration, solves a shortest path problem and then augments unit amount of flow on this path. The proposed algorithm is superior to Hu's algorithm in the following respects: (i) flow augmenting paths are obtained without solving shortest path problems and thereby saving computations; (ii) the augmented flow at each iteration can be more than 1 which is a desirable feature if the values of b_{ij}^k are large; and (iii) the algorithm can solve problems in which b_{ij}^k are noninteger or even irrational.

4.5 *DESCRIPTION OF THE ALGORITHM*

A formal statement of the CCNF algorithm is given below.

Step 1 : Let T be the tree of shortest paths rooted at source with c^1_{ij} as the length of each arc $(i,j) \in A$. Set $B^1 = T$, $U^0 = A-T$ and $B^2 = \ldots = B^p = U^1 = \ldots = U^p = \phi$. The initial optimum basis structure is $(B^1 \cup B^2 \cup \ldots \cup B^p, U^0, U^1, \ldots, U^p)$ with the associated flow $x_{ij} = 0$, $\forall (i,j) \in A$. Compute π_j by setting $\pi_s = 0$ and using $\pi_j - \pi_i = c^1_{ij}$, $\forall (i,j) \in B$. Set $\bar{v} = 0$, $\bar{Z} = 0$. Go to Step 2.

Step 2 : Let $\bar{P}$ and $\underline{P}$ be the sets of forward and backward arcs in the basic path P respectively. Define the number $\bar{w}_{ij}$ for each $(i,j) \in P$ as follows:

$$\bar{w}_{ij} = \begin{cases} b^k_{ij} - x_{ij}, & \text{if } (i,j) \in \bar{P} \cap B^k, \\ x_{ij} - b^{k-1}_{ij}, & \text{if } (i,j) \in \underline{P} \cap B^k. \end{cases}$$

Compute

$$\bar{w} = \min_{(i,j) \in P} \{\bar{w}_{ij}\}$$

and

$$\bar{y} = \min. \{\bar{w}, (v-\bar{v})\}.$$

Update x_{ij}, $\bar{v}$ and $\bar{Z}$ as follows:

$$x_{ij} = \begin{cases} x_{ij} + \bar{y}, & \forall (i,j) \in \bar{P}, \\ x_{ij} - \underline{y}, & \forall (i,j) \in \underline{P}, \\ x_{ij}, & \forall (i,j) \notin P, \end{cases}$$

$\bar{v} = \bar{v} + \bar{y}$,

$\bar{Z} = \bar{Z} + \pi_t \bar{y}$.

If $\bar{v} = v$, go to Step 5; otherwise go to Step 3.

Step 3 : Let (α,β) be the arc nearest to source among the arcs for which $\bar{w}_{ij} = \bar{w}$. Drop the arc (α,β) from the basis. Let T_s and T_t be the resulting subtrees containing source and sink respectively. Let Q be the cutset separating T_s and T_t. Let $\bar{Q}$ and $\underline{Q}$ be the sets of forward and backward arcs in Q respectively. Further, let
$\hat{U}^k = \{(i,j) \in U^0 \cup U^1 \cup \ldots \cup U^p \cup \{(\alpha,\beta)\} : x_{ij} = b^k_{ij}\}$,
$\forall\ k=0,1,\ldots,p$. Define the number $\mu_{h\ell}$ for each $(h,\ell) \in Q$ as follows:

$$\mu_{h\ell} = \begin{cases} \pi_h - \pi_\ell + c^{k+1}_{h\ell}, & \text{if } (h,\ell) \in \bar{Q} \cap \hat{U}^k, \\ \pi_\ell - \pi_h - c^k_{h\ell}, & \text{if } (h,\ell) \in \underline{Q} \cap \hat{U}^k. \end{cases}$$

Compute $\mu = \min\limits_{(h,\ell) \in Q} \{\mu_{h\ell}\}$. If $\mu = \infty$, go to Step 4; otherwise identify an arc (q,r) for which $\mu_{qr} = \mu$.
Let $(q,r) \in \hat{U}^f$. Obtain the new optimum basis structure by replacing (α,β) by (q,r) as a $(f+1)$-basic arc if $(q,r) \in \bar{Q}$, or as a f-basic arc if $(q,r) \in \underline{Q}$. Update π_j as follows:

$$\pi_j = \begin{cases} \pi_j , & \forall\ j \in T_s , \\ \pi_j + \mu, & \forall\ j \in T_t , \end{cases}$$

and go to Step 2.

Step 4 : The CCNF problem is infeasible. STOP.

Step 5 : The optimum flow of the CCNF problem is x_{ij} with the cost of flow $\bar{Z}$. STOP.

4.6 COMPLEXITY OF THE ALGORITHM

The following theorem establishes a tight bound on the number of computations performed by the algorithm when all b_{ij}^{k} are integer.

Theorem 4.2 : If all b_{ij}^{k} are integer, the complexity of the CCNF algorithm is $O(mnv)$.

Proof : Let us call an iteration of the CCNF algorithm as a nondegenerate iteration if positive amount of flow is augmented on the basic path in that iteration; otherwise it is called a degenerate iteration. Let us further define a nondegenerate subtree as a subtree containing source in which a positive amount of flow can be augmented from source to any of its nodes through the arc belonging to the subtree. The algorithm follows the rule that the arc nearest to the source among the flow blocking arcs leaves the basis. As a result, T_s is always a nondegenerate subtree. Each dual simplex iteration adds one arc to the nondegenerate subtree. Hence, after atmost $(n-1)$ degenerate iterations, the nondegenerate subtree includes the sink and there is a nondegenerate iteration. If all b_{ij}^{k} are integer, then at least a unit amount of flow is augmented in a nondegenerate iteration. The algorithm, thus, terminates in atmost $(n-1)v$ iterations. Since the number of computations performed at each iteration are $O(m)$, the complexity of the algorithm is $O(mnv)$.

Remark 4.1 : It is obvious from the above proof that the algorithm terminates in a finite number of iterations even if b_{ij}^k are not integers.

4.7 NUMERICAL EXAMPLE

In this section, we solve a numerical example to illustrate various steps of the CCNF algorithm. The network is shown in Fig. 4.1. Data of the problem for p=2 is given in Table 4.1. Nodes 1 and 6 are the source and sink nodes respectively. The flow to be transshipped from source to sink is 15 units.

The steps of the algorithm are summarized in Table 4.2. The desired flow is established in six iterations. In the table, ↓ indicates the arc leaving the basis and ↑ indicates the arc entering the basis. The optimum flow is given in the last row of the table. The basis obtained at each iteration is shown in Fig. 4.2. The optimum dual variables associated with the nodes are mentioned above them.

Table 4.1 : Data of the numerical example

(i,j)	b_{ij}^o	b_{ij}^1	b_{ij}^2	c_{ij}^1	c_{ij}^2
(1,2)	0	8	11	1	2
(1,3)	0	6	10	1	3
(2,3)	0	5	5	2	10
(2,4)	0	4	8	-1	1
(2,5)	0	3	5	3	4
(3,4)	0	5	8	2	5
(3,5)	0	3	9	-1	2
(5,4)	0	4	4	2	10
(4,6)	0	6	10	2	6
(5,6)	0	10	14	3	5

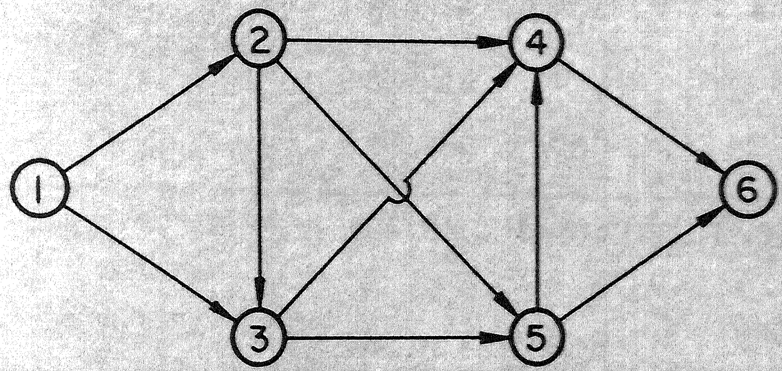

Fig. 4.1 Numerical example for the CCNF problem

Table 4.2 : Solution of the numerical example

Iteration No.	$\bar{v}$	$\bar{Z}$		Basic Arcs					Nonbasic Arcs				
1.	0	0	(i,j)	(1,2)	(1,3)	(2,4)	(3,5)	(4,6)	(2,3)	(2,5)	(3,4)	(5,4)	(5,6)
			x_{ij}	0	0	0	0	0	0	0	0	0	0
			Status	B^1	B^1	B^1	B^1	B^1	U^0	U^0	U^0	U^0	U^0
			$\bar{w}_{ij}$	8	-	4	-	5	-	-	-	-	-
			$\mu_{k\ell}$	-	-	2 ↓	-	-	-	-	3	2	1 ↑
2.	4	8	(i,j)	(1,2)	(1,3)	(5,6)	(3,5)	(4,6)	(2,3)	(2,5)	(3,4)	(5,4)	(2,4)
			x_{ij}	4	0	0	0	4	0	0	0	0	4
			Status	B^1	B^1	B^1	B^1	B^1	U^0	U^0	U^0	U^0	U^1
			$\bar{w}_{ij}$	-	6	10	3	-	-	-	-	-	-
			$\mu_{k\ell}$	-	-	-	3 ↓	-	-	4	2	-	1 ↑
3.	7	17	(i,j)	(1,2)	(1,3)	(5,6)	(2,4)	(4,6)	(2,3)	(2,5)	(3,4)	(5,4)	(3,5)
			x_{ij}	4	3	3	4	4	0	0	0	0	3
			Status	B^1	B^1	B^1	B^2	B^1	U^0	U^0	U^0	U^0	U^1
			$\bar{w}_{ij}$	4	-	-	4	2	-	-	-	-	-
			$\mu_{k\ell}$	-	-	-	-	4 ↓	-	3	-	-	2 ↑
4.	9	25	(i,j)	(1,2)	(1,3)	(5,6)	(2,4)	(3,5)	(2,3)	(2,5)	(3,4)	(5,4)	(4,6)
			x_{ij}	6	3	3	6	3	0	0	0	0	6
			Status	B^1	B^1	B^1	B^2	B^2	U^0	U^0	U^0	U^0	U^1
			$\bar{w}_{ij}$	-	3	7	-	6	-	-	-	-	-
			$\mu_{k\ell}$	-	2 ↓	-	-	-	2	1 ↑	-	-	2

66

Iteration No.	$\bar{v}$	$\bar{z}$		Basic Arcs					Nonbasic Arcs				
			(i,j)	(1,2)	(2,5)	(5,6)	(2,4)	(3,5)	(2,3)	(1,3)	(3,4)	(5,4)	(4,6)
			x_{ij}	0	0	6	6	6	0	6	0	0	6
5.	12	43	Status	B^1	B^1	B^1	B^2	B^2	U^o	U^1	U^o	U^o	U^1
			$\bar{w}_{ij}$	2	3	4	–	–	–	–	–	–	–
			$\mu_{k\ell}$	1	–	–	–	–	–	1	–	–	–
				↓						↑			
			(i,j)	(1,3)	(2,5)	(5,6)	(2,4)	(3,5)	(2,3)	(1,2)	(3,4)	(5,4)	(4,6)
			x_{ij}	6	2	8	6	6	0	8	0	0	6
6.	14	57	Status	B^2	B^1	B^1	B^2	B^2	U^o	U^1	U^o	U^o	U^1
			$\bar{w}_{ij}$	4	–	2	–	3	–	–	–	–	–
	15	65	(i,j)	(1,3)	(2,5)	(5,6)	(2,4)	(3,5)	(2,3)	(1,2)	(3,4)	(5,4)	(4,6)
			x_{ij}	7	2	9	6	7	0	8	0	0	6

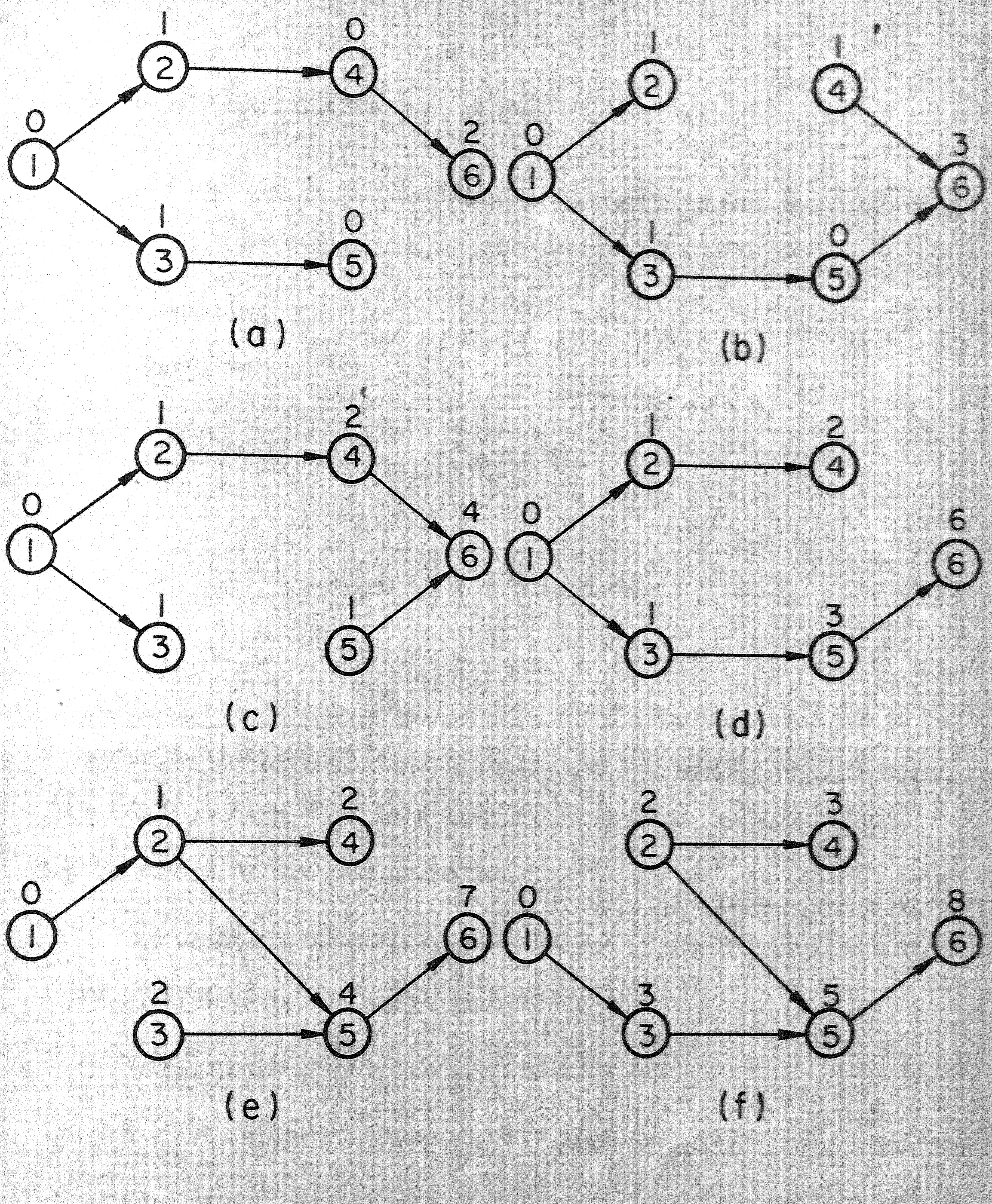

Fig. 4.2 Basis in various iterations

4.8 CONSTRAINED MAXIMUM FLOW PROBLEM

The problem to maximize flow in a network subject to a budgetary constraint, can be stated as

$$\text{Maximize} \quad v, \tag{4.40}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i \neq s,t, \quad \forall\, i\ \varepsilon\ N, \\ v, & \text{if } i=t, \end{cases} \tag{4.41}$$

$$0 \leq x_{ij} \leq b_{ij}\ , \ \forall\,(i,j)\ \varepsilon\ A, \tag{4.42}$$

$$\sum_{(i,j)\ \varepsilon\ A} c_{ij}\, x_{ij} \leq D. \tag{4.43}$$

We refer to the problem (4.40) - (4.43) as the <u>Constrained Maximum Flow</u> (CMF) problem. In this section, we show how the CMF problem can be solved by the CCNF algorithm.

The complementary slackness conditions of the CMF problem can be shown equivalent to the following conditions:

$$0 < x_{ij} < b_{ij} \Longrightarrow \sigma_i - \sigma_j = \mu c_{ij}, \ \forall\,(i,j)\ \varepsilon\ A, \tag{4.44}$$

$$\sigma_i - \sigma_j < \mu c_{ij} \Longrightarrow x_{ij} = 0 \quad , \ \forall\,(i,j)\ \varepsilon\ A, \tag{4.45}$$

$$\sigma_i - \sigma_j > \mu c_{ij} \Longrightarrow x_{ij} = b_{ij} \quad , \ \forall\,(i,j)\ \varepsilon\ A, \tag{4.46}$$

$$v > 0 \quad \Longrightarrow \quad \sigma_s - \sigma_t = 1, \tag{4.47}$$

$$\mu\,(D - \sum_{(i,j)\ \varepsilon\ A} c_{ij}\, x_{ij}) = 0, \tag{4.48}$$

where σ_j and μ are the dual variables associated with the constraints (4.41) and (4.43) respectively.

Substituting $\pi_j = -\sigma_j/\mu$ for each $j \in N$ in (4.44)-(4.48) we get the following equivalent conditions:

$$0 < x_{ij} < b_{ij} \Longrightarrow \pi_j - \pi_i = c_{ij} \quad , \forall (i,j) \in A, \tag{4.49}$$

$$\pi_j - \pi_i < c_{ij} \Longrightarrow x_{ij} = 0 \quad , \forall (i,j) \in A, \tag{4.50}$$

$$\pi_j - \pi_i > c_{ij} \Longrightarrow x_{ij} = b_{ij} \quad , \forall (i,j) \in A, \tag{4.51}$$

$$v > 0 \Longrightarrow \pi_t - \pi_s = 1/\mu \quad , \tag{4.52}$$

$$\mu(D - \sum_{(i,j) \in A} c_{ij} x_{ij}) = 0. \tag{4.53}$$

Consider the CCNF problem with $p=1$ and $b^1_{ij} = b_{ij}$ and $c^1_{ij} = c_{ij}$ for each $(i,j) \in A$. The conditions (4.49)-(4.51) are the complementary slackness conditions of the CCNF problem. The condition (4.52) can always be satisfied by suitably choosing μ. Hence the only additional condition is (4.53).

Consider the optimum solution of the CCNF problem when $Z = D$. This solution satisfies (4.43) and hence is an optimum solution of the CMF problem. However, if the maximum flow is established in the network before Z equals D, then the constraint (4.43) is not a binding constraint and this flow satisfies (4.53) with $\mu = 0$.

4.9 *TIME-COST TRADEOFF PROBLEM IN CPM NETWORKS*

Consider a project consisting of a number of activities. The precedence relations between these activities are indicated by identifying the activities with the arcs of a directed network $G = (N,A)$. Let

nodes s and t mark the initial and final events of the project respectively.

Each activity $(i,j) \varepsilon A$ has an associated normal completion time and a crash completion time. The cost of performing the activity between these two extreme times is given by a piecewise linear convex function which consists of q_{ij} linear segments. Associated with each activity $(i,j) \varepsilon A$ are numbers $a_{ij}^{1} > a_{ij}^{2} > \ldots > a_{ij}^{q_{ij}+1} \geq 0$, and the cost of shortening the completion time of the activity (i,j) in the interval $(a_{ij}^{k+1}, a_{ij}^{k})$ by one unit is b_{ij}^{k} such that $0 < b_{ij}^{1} < b_{ij}^{2} < \ldots < b_{ij}^{q_{ij}}$.

The problem of finding the minimum cost of shortening the project to a given duration λ is known as the <u>Time-Cost Tradeoff</u> (TCT) problem. The plot of the least project cost as a function of the project duration is known as the <u>project cost curve</u>. In this section, we show that the CCNF algorithm can be used to obtain the project cost curve.

Associate a variable ρ_i with each $i \varepsilon N$, which denotes the time at which event i starts. Let t_{ij}^{k} denote the shortening of activity (i,j) in the interval (a^{k+1}, a^{k}). Since the cost function is convex, shortening of the activity in different intervals can be treated independently. The TCT problem can be stated as follows:

$$\text{Minimize} \quad \sum_{(i,j)\,\varepsilon\, A} \sum_{k=1}^{q_{ij}} b_{ij}^{k}\, t_{ij}^{k} \,, \tag{4.54}$$

subject to

$$\rho_t - \rho_s \leq \lambda \,, \tag{4.55}$$

$$\rho_i - \rho_j + a_{ij}^1 - \sum_{k=1}^{q_{ij}} t_{ij}^k \leq 0, \ \forall (i,j) \in A, \tag{4.56}$$

$$0 \leq t_{ij}^k \leq a_{ij}^k - a_{ij}^{k+1} \ , \ \forall (i,j) \in A, \ \forall k=1,\ldots,q_{ij} \ . \tag{4.57}$$

Associate the variables v, x_{ij} and δ_{ij}^k with the constraints (4.55), (4.56) and (4.57) respectively. The dual of the TCT problem is as follows:

$$\text{Maximize} \quad \sum_{(i,j) \in A} a_{ij}^1 x_{ij} - \sum_{(i,j) \in A} \sum_{k=1}^{q_{ij}} (a_{ij}^k - a_{ij}^{k+1}) \ \delta_{ij}^k - \lambda v, \tag{4.58}$$

subject to

$$\sum_{(j,i) \in I(i)} x_{ji} - \sum_{(i,j) \in O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0 \ , & \text{if } i \neq s,t, \ \forall i \in N, \\ v \ , & \text{if } i=t, \end{cases} \tag{4.59}$$

$$x_{ij} - \delta_{ij}^k \leq b_{ij}^k, \ \forall (i,j) \in A, \ \forall k=1,\ldots,q_{ij} \ , \tag{4.60}$$

$$v \geq 0; \ x_{ij} \geq 0, \ \forall (i,j) \in A; \text{ and } \delta_{ij}^k \geq 0, \ \forall (i,j) \in A, \ \forall k=1,\ldots,q_{ij} \ . \tag{4.61}$$

From $a_{ij}^1 > a_{ij}^2 > \ldots > a_{ij}^{q_{ij}+1}$ and (4.60), it follows that any optimum solution of (4.58)-(4.61) satisfies the following condition:

$$\delta_{ij}^k = \max. \{0, x_{ij} - b_{ij}^k \}, \ \forall (i,j) \in A, \ \forall k=1,\ldots,q_{ij}. \tag{4.62}$$

We can substitute (4.62) in (4.58) and eliminate (4.60) from the constraints. We then get the following equivalent problem:

$$\text{Maximize } \sum_{(i,j)\in A} a^1_{ij}x_{ij} - \sum_{(i,j)\,\varepsilon\, A}\ \sum_{k=1}^{q_{ij}} (a^k_{ij}-a^{k+1}_{ij}) \text{ max. } \{0, x_{ij}-b^k_{ij}\} - \lambda v, \tag{4.63}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = \begin{cases} -v\ , \text{ if } i=s, \\ 0\ , \text{ if } i\neq s,t, \ \forall\, i\,\varepsilon\, N, \\ v\ , \text{ if } i=t, \end{cases} \tag{4.64}$$

$$v \geq 0;\ x_{ij} \geq 0,\ \forall\,(i,j)\ \varepsilon\ A. \tag{4.65}$$

Let

$$p_{ij} = q_{ij} + 1 \quad ,\ \forall\,(i,j)\ \varepsilon\ A, \tag{4.66}$$

$$c^k_{ij} = -a^k_{ij},\ \forall\,(i,j)\ \varepsilon\ A,\ \forall\ k=1,\ldots,p_{ij}\ , \tag{4.67}$$

$$b^0_{ij} = 0 \text{ and } b^{p_{ij}}_{ij} = \infty,\ \forall\,(i,j)\ \varepsilon\ A. \tag{4.68}$$

Further, add an arc (t,s) to the network with unbounded capacity and cost of flow λ. We then get the following equivalent flow problem:

$$\text{Maximize } -Y(\lambda) = -\sum_{(i,j)\varepsilon A} C_{ij}(x_{ij}) - \lambda\, x_{ts}\ , \tag{4.69}$$

or, equivalently,

$$\text{Minimize } Y(\lambda) = \sum_{(i,j)\,\varepsilon\, A} C_{ij}(x_{ij}) + \lambda\, x_{ts}\ , \tag{4.70}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} x_{ji} - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = 0,\ \forall\, i\ \varepsilon\ N, \tag{4.71}$$

$$x_{ij} \geq 0,\ \forall\,(i,j)\ \varepsilon\ A, \tag{4.72}$$

where $C_{ij}(x_{ij})$ is the piecewise linear convex function as defined in Section 4.2. We refer to the problem (4.70)-(4.72) as the <u>Modified Time-Cost Tradeoff</u> (MTCT) problem.

Remark 4.2 : The MTCT problem is not the dual of the TCT problem; however, both have the same optimum solution. The optimum value of the MTCT problem is to be multiplied by -1 to obtain the optimum value of the TCT problem.

The project cost curve is obtained by plotting $-Y(\lambda)$ as λ is varied from the normal duration of the project to the crash duration of the project. We now show how the CCNF algorithm obtains the project cost curve.

Let us partition A into the sets $B^1, B^2, \ldots, B^p, U^o, U^1, \ldots, U^p$ and define a feasible basis structure of the MTCT problem as $(B^1 \cup B^2 \cup \ldots \cup B^p \cup \{(t,s)\},\ U^o, U^1, \ldots, U^p)$, where $B^1 \cup B^2 \cup \ldots \cup B^p \cup \{(t,s)\}$ is a spanning tree, and B^k and U^k satisfy the flow restrictions (4.5) and (4.4) respectively. It may be noted that the basis of the MTCT problem consists of two subtrees, which contain source and sink respectively, joined by the arc (t,s). It can be easily shown that the necessary and sufficient conditions for a feasible basis structure of the MTCT problem to be an optimum basis structure are that there exist numbers π_j satisfying the following conditions:

$$\lambda = -\pi_t + \pi_s \,, \tag{4.73}$$

$$\pi_j - \pi_i = c_{ij}^k, \ \forall (i,j) \in B^k, \ \forall k=1,\ldots,p, \tag{4.74}$$

$$c_{ij}^k \le \pi_j - \pi_i \le c_{ij}^{k+1} \,, \ \forall (i,j) \in U^k, \ \forall k=0,\ldots,p. \tag{4.75}$$

We refer to the conditions (4.73)-(4.75) as the optimality conditions for the MTCT problem. Observe that the conditions (4.74)

and (4.75) are the optimality conditions of the CCNF problem. We refer to π_j as the optimum dual variables of the MTCT problem. It follows from Remark 4.2 that if ρ_j are the optimum even start times of the TCT problem, then $\rho_j = -\pi_j$ for all $j \in N$. Without any loss of generality, we assume in the subsequent discussion that $\pi_s = 0$.

We first note that any CPM network is acyclic and hence Assumption 4.1 is satisfied. We also note that $b_{ij}^p = \infty$ for each $(i,j) \in A$, which implies that maximum flow in the network is unbounded.

Let $(\underline{v}, \bar{v})$ be one of the characteristic intervals of the CCNF problem and the associated optimum basis structure be $(B^1 \cup B^2 \cup \ldots \cup B^p, U^o, U^1, \ldots, U^p)$ with the optimum dual variables π_j. Let $(\bar{B}^1 \cup \bar{B}^2 \cup \ldots \cup \bar{B}^p, \bar{U}^o, \bar{U}^1, \ldots, \bar{U}^p)$ be the alternate optimum basis structure for $v = \bar{v}$ with the associated dual variables σ_j. It follows from (4.37) that

$$\sigma_j = \begin{cases} \pi_j & , \ \forall\, j \in T_s, \\ \pi_j + \mu, & \forall\, j \in T_t \ . \end{cases} \tag{4.76}$$

Let $\bar{\lambda} = -\pi_t$ and $\underline{\lambda} = -\sigma_t$. Clearly $\bar{\lambda} \geq \underline{\lambda}$.

Let $\hat{B}^k$ and $\hat{U}^k$ be defined by (4.32) and (4.31) respectively. It is easy to see that $(\hat{B}^1 \cup \hat{B}^2 \cup \ldots \cup \hat{B}^p \cup \{(t,s)\}, \hat{U}^o, \hat{U}^1, \ldots, \hat{U}^p)$ is the optimum basis structure of the MTCT problem for $\lambda = \bar{\lambda}$ with the associated optimum dual variables π_j. It is also easy to see that the same basis structure is the optimum basis structure of MTCT problem for $\lambda = \underline{\lambda}$ with the associated dual variables σ_j. It is now obvious

that $(\hat{B}^1 \cup \hat{B}^2 \cup \ldots \cup \hat{B}^p \cup \{(t,s)\}, \hat{U}^o, \hat{U}^1, \ldots, \hat{U}^p)$ is the optimum basis structure of the MTCT problem for all $\lambda \in (\underline{\lambda}, \bar{\lambda})$ with the associated optimum dual variables $\pi'_j = \theta\,\sigma_j + (1-\theta)\,\pi_j$, where $\theta = (\bar{\lambda}-\lambda)/(\bar{\lambda}-\underline{\lambda})$. Hence, the optimum completion time for each $(i,j) \in A$ in the interval $(\underline{\lambda}, \bar{\lambda})$ is $(\pi'_i - \pi'_j)$ and the slope of $-Y(\lambda)$ is $-x_{ts} = -\bar{v}$.

Now, consider the optimum basis structure of the CCNF problem for $v = \bar{v}$. The amount of flow which can be augmented on the basic path, $\bar{w}$, can be finite or infinite. If $\bar{w}$ is infinite, then it means that all the arcs in the basic path are forward arcs and their completion times are reduced to the crash completion times. Hence, the project duration is reduced to the crash duration and the TCT problem is infeasible for $\lambda < \underline{\lambda}$. However, if $\bar{w}$ is finite, then dropping the flow blocking arc yields an optimum basis structure of the MTCT problem for $\lambda = \underline{\lambda}$, which may allow further decrease in λ. Thus, we see that performing the dual simplex iteration for the CCNF problem corresponds to finding the characteristic interval for the MTCT problem and, conversely, finding the characteristic interval for the CCNF problem is equivalent to performing the simplex iteration for the MTCT problem.

We observed that for all $\lambda \in (\underline{\lambda}, \bar{\lambda})$, the optimum completion time of activity (i,j) is given by $(\pi'_i - \pi'_j)$ where $\pi'_j = \theta\,\sigma_j + (1-\theta)\,\pi_j$ and $\theta = (\bar{\lambda}-\lambda)/(\bar{\lambda}-\underline{\lambda})$. Substitution from (4.76) yields the following expression for $(\pi'_i - \pi'_j)$:

$$\pi_i' - \pi_j' = \begin{cases} (\pi_i - \pi_j) - \theta\,\mu, & \forall\,(i,j) \in \bar{Q}, \\ (\pi_i - \pi_j) + \theta\,\mu, & \forall\,(i,j) \in \underline{Q}, \\ (\pi_i - \pi_j) & , \forall\,(i,j) \notin Q. \end{cases} \tag{4.77}$$

As λ varies from $\bar{\lambda}$ to $\underline{\lambda}$, θ varies from 0 to 1. Accordingly, the optimum completion times of all activities belonging to Q decrease by an amount $\theta\,\mu$; these times for all activities belonging to $\underline{Q}$ increase by an amount $\theta\,\mu$; whereas the optimum completion times for other activities remain unchanged. The algorithm, thus, identifies a cutset at each iteration and completion times of all the activities belonging to this cutset are modified so that the project duration is decreased using minimum additional cost.

Phillips and Dessouky's algorithm [78] for the TCT problem defines appropriate lower and upper bounds for arcs at each iteration, identifies a cutset by applying a cut search procedure and modifies the completion times of all activities belonging to this cutset. The proposed algorithm is superior to the Phillips and Dessouky's algorithm in the following respects: (i) cutsets are implicitly enumerated by the algorithm and thereby enhacing its efficiency; and (ii) more general cost functions are considered.

Fulkerson [40] and Kelley [58] have proposed labelling procedures for the TCT problem which handle piecewise linear convex functions by introducing one arc for each linear segment. The algorithm proposed by us avoids this enlarging of the network which is highly desirable from computational point of view.

4.10 CONCAVE GAIN MAXIMUM FLOW PROBLEM

Given the input flow, the maximum flow with gains problem determines the maximum output flow when arcs have positive linear gains [47]. In this section, we consider a generalization of this problem where arcs can have piecewise linear concave gain functions. Mathematically, this problem can be stated as

$$\text{Maximize} \quad \beta , \tag{4.78}$$

subject to

$$\sum_{(j,i)\,\varepsilon\, I(i)} F_{ji}(x_{ji}) - \sum_{(i,j)\,\varepsilon\, O(i)} x_{ij} = \begin{cases} -\alpha, & \text{if } i=s, \\ 0, & \text{if } i\neq s,t, \quad \forall\, i \in N, \\ \beta, & \text{if } i=t, \end{cases} \tag{4.79}$$

$$x_{ij} \geq 0 , \quad \forall\, (i,j) \,\varepsilon\, A, \tag{4.80}$$

where $F_{ij}(x_{ij})$ is a piecewise linear concave function consisting of $p(i,j)$ linear segments. We refer to this problem as the <u>Concave Gain Maximum Flow</u> (CGMF) problem.

Let $p = \max\limits_{(i,j)\,\varepsilon A} \{p(i,j)\}$. Associated with each arc $(i,j)\ \varepsilon\ A$ are numbers $0 = b^0_{ij} < b^1_{ij} < \ldots < b^{p(i,j)}_{ij} = b^{p(i,j)+1}_{ij} = \ldots = b^p_{ij}$ and slope of $F_{ij}(x_{ij})$ in the interval (b^{k-1}_{ij}, b^k_{ij}) is r^k_{ij} such that $r^1_{ij} > r^2_{ij} > \ldots > r^p_{ij} > 0$.

Let us introduce the variables $y^1_{ij}, y^2_{ij}, \ldots, y^p_{ij}$ for each $(i,j)\ \varepsilon\ A$ and replace each x_{ij} by $\sum_{k=1}^{p} y^k_{ij}$ in the CGMF problem. We then get the following problem:

Maximize β ,

subject to $$\tag{4.81}$$

$$\sum_{(j,i)\,\varepsilon\, I(i)} \sum_{k=1}^{p} r_{ji}^{k}\, y_{ji}^{k} - \sum_{(i,j)\,\varepsilon\, O(i)} \sum_{k=1}^{p} y_{ij}^{k} = \begin{cases} -\alpha, & \text{if } i=s, \\ 0, & \text{if } i\neq s,t, \ \forall\, i\,\varepsilon\, N, \\ \beta, & \text{if } i=t, \end{cases} \tag{4.82}$$

$$0 \leq y_{ij}^{k} \leq b_{ij}^{k} - b_{ij}^{k-1}, \ \forall\, (i,j)\ \varepsilon\ A, \ \forall\ k=1,\ldots,p. \tag{4.83}$$

It is easy to see that (4.81)-(4.83) is equivalent to the CGMF problem if its every optimum solution satisfies the following conditions:

$$y_{ij}^{k} > 0 \Longrightarrow y_{ij}^{\ell} = b_{ij}^{\ell} - b_{ij}^{\ell-1}, \ \forall\ \ell = 1,\ldots, k-1, \tag{4.84}$$

$$y_{ij}^{k} < b_{ij}^{k} - b_{ij}^{k-1} \Longrightarrow y_{ij}^{\ell} = 0 \ , \ \forall\ \ell = k+1,\ldots,p. \tag{4.85}$$

We now show that these conditions are indeed satisfied. The dual of (4.81)-(4.83) is

$$\text{Minimize} \quad -\alpha\rho_s + \sum_{(i,j)\,\varepsilon\, A} \sum_{k=1}^{p} (b_{ij}^{k} - b_{ij}^{k-1})\, \delta_{ij}^{k} , \tag{4.86}$$

subject to

$$\rho_j\, r_{ij}^{k} - \rho_i + \delta_{ij}^{k} \geq 0, \ \forall\, (i,j)\ \varepsilon\ A, \ \forall\ k=1,\ldots,p, \tag{4.87}$$

$$-\rho_t \geq 1, \tag{4.88}$$

$$\delta_{ij}^{k} \geq 0, \ \forall\, (i,j)\ \varepsilon\ A, \ \forall\ k=1,\ldots,p, \tag{4.89}$$

where ρ_i and δ_{ij}^{k} are the dual variables associated with (4.82) and (4.83) respectively.

The complementary slackness conditions of (4.81)-(4.83) are

$$\delta_{ij}^{k}(b_{ij}^{k}-b_{ij}^{k-1}-y_{ij}^{k}) = 0, \ \forall (i,j) \in A, \ \forall k=1,\dots,p, \tag{4.90}$$

$$y_{ij}^{k}(\rho_j r_{ij}^{k}-\rho_i+\delta_{ij}^{k}) = 0, \ \forall (i,j) \in A, \ \forall k=1,\dots,p, \tag{4.91}$$

$$\beta(-\rho_t-1) = 0, \tag{4.92}$$

which can be equivalently stated as follows:

$$0 < y_{ij}^{k} < b_{ij}^{k} - b_{ij}^{k-1} \implies \rho_i = r_{ij}^{k}\rho_j, \ \forall (i,j) \in A, \ \forall k=1,\dots,p, \tag{4.93}$$

$$r_{ij}^{k} \quad \rho_j > \rho_i \implies y_{ij}^{k} = 0 \quad , \ \forall (i,j) \in A, \ \forall k=1,\dots,p, \tag{4.94}$$

$$r_{ij}^{k} \quad \rho_j < \rho_i \implies y_{ij}^{k} = b_{ij}^{k} - b_{ij}^{k-1}, \ \forall (i,j) \in A, \ \forall k=1,\dots,p, \tag{4.95}$$

$$\beta > 0 \implies \rho_t = -1. \tag{4.96}$$

Theorem 4.3 : Every optimum solution of (4.81)-(4.83) satisfies (4.84) and (4.85).

Proof : Suppose that the theorem is not true and there exists an arc (i,j) for which $y_{ij}^{k} > 0$ and $y_{ij}^{\ell} < b_{ij}^{\ell} - b_{ij}^{\ell-1}$ such that $\ell < k$. Since all $r_{ij}^{k} > 0$, it follows from (4.93) and (4.96) that $\rho_j < 0$ for all $j \in N$. It further follows from (4.91) and (4.90) that

$$y_{ij}^{k} > 0 \implies \rho_j r_{ij}^{k} - \rho_i + \delta_{ij}^{k} = 0, \tag{4.97}$$

and

$$y_{ij}^{\ell} < b_{ij}^{\ell} - b_{ij}^{\ell-1} \implies \delta_{ij}^{\ell} = 0. \tag{4.98}$$

Using (4.87) and (4.98) we get

$$y_{ij}^{\ell} < b_{ij}^{\ell} - b_{ij}^{\ell-1} \implies \rho_j r_{ij}^{\ell} - \rho_i \geq 0 . \tag{4.99}$$

Hence

$$y_{ij}^{k} > 0 \quad \text{and} \quad y_{ij}^{\ell} < b_{ij}^{\ell} - b_{ij}^{\ell-1} \implies \rho_j (r_{ij}^{\ell} - r_{ij}^{k}) - \delta_{ij}^{k} \geq 0. \tag{4.100}$$

Since $\rho_j < 0$ and $\delta_{ij}^{k} \geq 0$, (4.100) implies that $r_{ij}^{\ell} \leq r_{ij}^{k}$ which contradicts our assumption that $r_{ij}^{\ell} > r_{ij}^{k}$.

Therefore, the problem (4.81) - (4.83) is equivalent to the CGMF problem. This equivalence suggests that the CGMF problem can be solved as the maximum flow problem with linear gains by introducing one arc for each linear segment. This transformation enlarges the network substantially. We now suggest modifications in the CCNF algorithm in order to solve the CGMF problem which does not require any additional arc to be introduced. We first employ logarithmic transformation to simplify (4.93)-(4.96).

Let

$$\begin{aligned} \pi_i &= \log(-\rho_i), \ \forall\, i \in N, \text{ and} \\ c_{ij}^{k} &= \log(1/r_{ij}^{k}), \ \forall\, (i,j) \in A, \ \forall\, k = 1,\dots,p. \end{aligned} \tag{4.101}$$

Note that $c_{ij}^{1} < c_{ij}^{2} < \dots < c_{ij}^{p}$ for all $(i,j) \in A$. Substituting (4.101) in (4.93)-(4.96), we get the following conditions:

$$0 < y_{ij}^{k} < b_{ij}^{k} - b_{ij}^{k-1} \implies \pi_j - \pi_i = c_{ij}^{k}, \ \forall\, (i,j) \in A, \ \forall\, k=1,\dots,p, \tag{4.102}$$

$$\pi_j - \pi_i < c_{ij}^{k} \implies y_{ij}^{k} = 0, \ \forall\, (i,j) \in A, \ \forall\, k = 1,\dots,p, \tag{4.103}$$

$$\pi_j - \pi_i > c_{ij}^{k} \implies y_{ij}^{k} = b_{ij}^{k} - b_{ij}^{k-1}, \ \forall\, (i,j) \in A, \ \forall\, k=1,\dots,p, \tag{4.104}$$

$$\beta > 0 \implies \pi_t = 0. \tag{4.105}$$

The conditions (4.102)-(4.104) are the complementary slackness conditions of the CCNF problem. The additional conditions (4.105) can always be satisfied, as we show later. Therefore, we find that the complementary slackness conditions of the CCNF problem and the CGMF problem are same. This is a strong relationship between the two problems.

Assumption 4.1 in the context of the CGMF problem implies that the network does not contain any flow generating directed cycle with r_{ij}^{1} as the gain of each arc $(i,j) \varepsilon A$. In this case, Grinold [47] has shown that any optimum basis of the problem is a spanning tree. Hence, the concepts of feasible and optimum basis structures for the CCNF problem remain valid for the CGMF problem. This observation permits us to develop the CGMF algorithm, similar to the CCNF algorithm.

The CGMF algorithm treats α as a parameter. The initial optimum basis structure of the CGMF problem is obtained by solving a shortest path problem. The associated optimum dual variables are obtained by setting $\pi_t = 0$ and then using (4.21). Finding the characteristic interval for this basis structure is slightly involved due to the gain functions. The labelling procedure described by Grinold [47] to augment the flow from source to sink can be modified for our problem. Calculations in the dual simplex iteration remain unchanged, except that the dual variables are updated by substracting μ from π_j for all $j \varepsilon T_s$.

Intuitively speaking, the CGMF algorithm augments the flow from source to sink on the paths with maximum gains which are implicitly enumerated by the algorithm. The finiteness of the CGMF algorithm follows from Remark 4.1.

4.11 CONVEX COST CAPACITY EXPANSION MAXIMUM FLOW PROBLEM

The problem of optimally allocating a prescribed budget D to increase the capacities of various arcs, in order to maximize the flow in a network, is of considerable practical significance. In Section 3.8, this problem with linear capacity expansion costs was considered. In this section, we show that when capacity expansion costs are given by piecewise linear convex functions, this problem can be solved by the CCNF algorithm. The mathematical formulation of this problem is as follows:

$$\text{Maximize} \quad v, \tag{4.106}$$

subject to

$$\sum_{(j,i) \in I(i)} x_{ji} - \sum_{(i,j) \in O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i \neq s,t, \ \forall\, i \in N, \\ v, & \text{if } i=t \end{cases} \tag{4.107}$$

$$x_{ij} \leq b_{ij} + y_{ij}, \ \forall\, (i,j) \in A, \tag{4.108}$$

$$\sum_{(i,j) \in A} D_{ij}\,(y_{ij}) \leq D, \tag{4.109}$$

$$x_{ij} \geq 0 \quad \text{and} \quad y_{ij} \geq 0, \ \forall\, (i,j) \in A, \tag{4.110}$$

where $D_{ij}(x_{ij})$ is a piecewise linear convex function with positive slopes. We assume that $D_{ij}(0) = 0$ for each $(i,j) \in A$. We refer

to the problem (4.106)-(4.110) as the <u>Convex Cost Capacity Expansion Maximum Flow</u> (CCCEMF) problem.

We assume that the constraint (4.109) is a binding constraint. If this constraint is not binding, then the CCCEMF problem is equivalent to the maximum flow problem where y_{ij} are at their upper permissible bounds.

It follows from the above assumption and (4.108) that the optimum solution of the CCCEMF problem satisfies the following conditions:

$$y_{ij} = \max.\ \{0,\ x_{ij}-b_{ij}\},\ \forall\ (i,j)\ \varepsilon\ A. \tag{4.111}$$

We can substitute (4.111) in (4.109) and eliminate y_{ij}. We then get the following equivalent problem:

$$\text{Maximize}\quad v, \tag{4.112}$$

subject to

$$\sum_{(j,i)\ \varepsilon\ I(i)} x_{ji} - \sum_{(i,j)\ \varepsilon\ O(i)} x_{ij} = \begin{cases} -v, & \text{if } i=s, \\ 0, & \text{if } i\neq s,t,\ \forall\ i\ \varepsilon\ N, \\ v, & \text{if } i=t, \end{cases} \tag{4.113}$$

$$\sum_{(i,j)\ \varepsilon\ A} C_{ij}(x_{ij}) \leq D, \tag{4.114}$$

$$x_{ij} \geq 0,\ \forall\ (i,j)\ \varepsilon\ A, \tag{4.115}$$

where $C_{ij}(x_{ij})$ is the function obtained by translating $D_{ij}(y_{ij})$ by b_{ij} units.

The following theorem establishes a close relationship between the optimum solutions of the CCCEMF problem and the CCNF problem with $C_{ij}(x_{ij})$ as the cost function for arc $(i,j)\ \varepsilon\ A$.

<u>Theorem 4.4</u> : Let x_{ij}^{o} be an optimum solution of the CCNF problem for $v = v^{o}$ with Z^{o} as the value of objective function. Then, x_{ij}^{o} is an optimum solution of the CCCEMF problem for $D = Z^{o}$ with v^{o} as the value of objective function.

<u>Proof</u> : Let x_{ij}^{*} be the optimum solution of the CCCEMF problem for $D = Z^{o}$ with the value of objective function v^{*}. Let $Z^{*} = \sum_{(i,j) \in A} C_{ij}(x_{ij}^{*})$. We know that Z^{o} is the minimum cost of shipping v^{o} amount of flow from source to sink. We also know that every additional flow from source to sink requires additional cost because $C_{ij}(x_{ij})$ has nonnegative slopes for each $(i,j) \in A$. Thus $v^{*} > v^{o}$ implies $Z^{*} > Z^{o}$ which contradicts (4.109). Hence $v^{*} \leq v^{o}$ and $Z^{*} \leq Z^{o}$. Since x_{ij}^{o} is also a feasible solution of the CCCEMF problem for $D = Z^{o}$, we must have $v^{*} = v^{o}$. Further, $Z^{*} = Z^{o}$ because x_{ij}^{*} is also a feasible solution of the CCNF problem for $v = v^{o}$ and $Z^{*} < Z^{o}$ contradicts the optimality of x_{ij}^{o}. Therefore, $v^{o} = v^{*}$ and $Z^{o} = Z^{*}$ and x_{ij}^{o} is an optimum solution of the CCCEMF problem for $D = Z^{o}$.

This theorem shows that the CCNF algorithm also solves the CCCEMF problem. The CCNF algorithm then can be viewed as treating D as a parameter. It starts with $D = 0$ and increases its value in subsequent iterations until either the prescribed value of D is attained or the maximum flow is established in the network.

4.12 COMPUTATIONAL RESULTS

A computer program, named PCCNF, was written in FORTRAN_IV for the CCNF algorithm. The PCCNF stores the network in the form of an arc list and computations are performed by using a minor variant of the augmented predecessor index method described in Murty [72]. The listing of the program is given in the Appendix. The program was debugged and tested on DEC-1090 computer system. The program occupies $15n + 5m + 2mp$ words of central memory.

We solved a number of network problems by PCCNF for various values of m,n and p and noted down the computational times. We also observed the effect on computational times of variations in the process by which b_{ij}^{k} and c_{ij}^{k} were generated. In this section , the results of these experiments are presented.

The computational experiments require a large number of randomly generated networks. Initially, we tried to use the program NETGEN- a wellknown network generator [96]. However, our code of NETGEN could not generate correct networks and we were unable to find the bugs. Hence we wrote our own network generator NG having similar characteristics as those of NETGEN. The program NG first generates a skeleton network and then adds arcs to it until specifications are met. In this manner, quite well structured networks are generated.

The data for the arcs was generated in the following manner:

$$b_{ij}^{o} = 0,$$

$$b_{ij}^{k+1} = b_{ij}^{k} + \text{IRAN}\ (10,30),\ \forall\, k = 1,\ldots,p,$$

$$c_{ij}^{1} = \text{IRAN}\ (10,\ 20),$$

$$c_{ij}^{k+1} = c_{ij}^{k} + \text{IRAN}\ (2,10),\ \forall\, k = 2,\ldots,p,$$

where IRAN (A,B) is a function that generates a uniformly distributed integer number in the interval (A,B).

Network problems were generated by NG for various values of the parameters: n, m and p. The parameters n, m and p were respectively varied from 20 to 200, 30 to 2000 and 1 to 5. For each set of parameters two problems were solved and average execution time to establish maximum flow at minimum cost was noted (the execution time is the CPU time exclusive of input and output times). These times are presented in Table 4.3.

The computational times demonstrate the efficiency of the GCNF algorithm for solving large-sized problems. Problems with $n = 200$, $m = 2000$ and $p = 5$ are solved by PGCNF in less than 37 seconds. We observe that if any two parameters are fixed, the computational times vary almost linearly with the third parameter. In fact, the approximate computational time T can be calculated by using the following formula:

$$T = 2nmp \times 10^{-5} \text{ seconds.}$$

We also varied the manner in which b_{ij}^{k} and c_{ij}^{k} were generated and observed its effect on the computational times. Some of these

Table 4.3 : Computational times of PCGNF (Execution times in seconds in DEC-1090)

No. of nodes	No. of arcs	No. of segments				
		1	2	3	4	5
	30	0.01	0.02	0.03	0.05	0.06
	50	0.02	0.05	0.06	0.08	0.12
20	100	0.05	0.08	0.11	0.16	0.22
	150	0.06	0.11	0.16	0.20	0.34
	200	0.06	0.15	0.23	0.33	0.45
	100	0.07	0.16	0.25	0.34	0.42
	200	0.22	0.45	0.70	0.85	1.26
50	500	0.53	0.91	1.49	1.80	2.55
	800	0.81	1.49	2.41	3.14	4.08
	1000	1.13	2.15	3.35	4.42	5.83
	200	0.28	0.59	0.87	1.22	1.39
	500	0.95	1.77	2.58	4.22	4.71
100	1000	1.75	3.28	6.18	8.07	9.79
	1500	3.56	5.86	8.07	10.23	14.18
	2000	4.36	8.51	13.49	17.00	22.03
	500	1.84	3.66	5.70	8.01	9.25
200	1000	4.43	8.73	9.95	17.27	18.89
	2000	8.21	14.64	20.56	27.16	36.75

observations are given below.

(i) When c_{ij}^{k} are generated by using $c_{ij}^{k+1} = c_{ij}^{k}$ + IRAN (2,X), and if X is increased from 10 to 100, the computation time increases by 20%.

(ii) When b_{ij}^{k} are generated by using $b_{ij}^{k+1} = b_{ij}^{k}$ + IRAN (10, X), the computational times decrease as X is increased. However, this change is not as pronounced as in the case of c_{ij}^{k}.

CHAPTER V

CONCLUDING REMARKS

In this dissertation, we studied the role of parametric programming in solving several generalizations of some of the wellknown network problems. We first studied the parametric analysis of some network problems and then considered the following generalizations:

(i) the parametric network feasibility problem;

(ii) the constrained network capacity expansion problem;

(iii) the parametric network capacity expansion problem;

(iv) the convex cost network flow problem;

(v) the time-cost tradeoff problem in CPM network;

(vi) the concave gain maximum flow problem; and

(vii) the convex cost capacity expansion maximum flow problem.

A common methodology based on parametric programming is suggested to solve these problems. This methodology utilizes the concept of optimum basis structure. The basis structure is a generalization of the working basis for bounded variable linear programs. Optimality is maintained because the optimum solution satisfies certain properties which are not satisfied by an arbitrary feasible solution. This is particularly true for convex functions where these properties permit us to consider several variables implicitly. Since in most of the problems considered by us, convex functions appear either in the constraints or in

the objective function, maintaining the optimality is very useful. Therefore, parametric programming is used to parametrize one parameter in order to find the optimum solution for the desired value of the parameter.

As indicated above, the optimum basis structure considers several variables implicitly. This, in certain problems, results in implicit consideration of several constraints. Hence the size of the optimum basis is reduced considerably. On account of this property, basis triangularity is preserved in most of the problems. This, in turn, yields algorithms which are intuitively appealing and easy to implement. Integral computations is also a byproduct of this property.

Another advantage of the parametric approach is the availability of additional information which may be of considerable managerial use. Algorithms based on this approach yield the optimum solution for all values of the parameter contained in an interval. In planning new product introduction or exploring expansion possibilities, this information will help in decision making. Also, the parametric approach easily lends itself to the sensitivity analysis.

There are several natural generalizations of the problems considered by us. A linear programming problem with convex cost functions is one such generalization. The concepts developed for the convex cost network flow problem may not be directly applicable to this problem. Some difficulties are encountered in selecting the parameter to be parametrized and finding the initial optimum basis structure. However, the concept of the feasible basis structure will still be useful. It may be possible

to develop a generalization of the simplex method which moves from one feasible basis structure to the another until an optimum basis structure is obtained.

In most of the problems considered by us, convex functions appear in the constraints or in the objective function. Nevertheless, our approach can be extended to solve problems where convex problems appear in the constraints as well as the objective function. Linear programming extensions of these problems can also be handled. The following allied generalizations can also be solved by suitable modifications and generalizations of the algorithms proposed in this dissertation:

(i) Parametric analysis of the maximum flow problem with gains, the maximum capacity path problem and the minimum ratio network problems.

(ii) The parametric network feasibility problem with nonlinear supply, demand and capacity functions.

(iii) The minimum cost flow problem with two additional linear constraints.

(iv) The parametric network capacity expansion problem with convex expansion costs.

REFERENCES

[1] Ahuja, R.K. and Mittal, A.K., "Maximal Arc-Disjoint and Node-Disjoint Flows in Two-Commodity Networks", to appear in Opsearch.

[2] Ahuja, R.K., Batra, J.L. and Gupta, S.K., "The Constrained Maximum Flow Problem," to appear in Scientific Management of Transport Systems, edited by N.K. Jaiswal, North Holland Publishing Co., Amsterdam, Holland, 1981.

[3] Aneja, Y.P. and Nair, K.P.K., "Bicriteria Transportation Problems," Management Science, 25 (1979), 73-78.

[4] Bacopoulus, A. and Singer, I., "On Convex Vectorial Optimization," Journal of Optimization Theory and Applications, 21 (1977), 175-188.

[5] Balachandran, V. and Thompson, G.L., "An Operator Theory of Parametric Programming for the Generalized Transportation Problem-I: Basic Theory," Naval Research Logistics Quarterly, 22 (1975), 79-100.

[6] Balachandran, V. and Thompson, G.L., "An Operator Theory of Parametric Programming for the Generalized Transportation Problem-II: Rim, Cost and Bound Operators," Naval Research Logistics Quarterly, 22 (1975), 101-125.

[7] Balachandran, V. and Thompson, G.L., "An Operator Theory of Parametric Programming for the Generalized Transportation Problem-III: Weight Operators", Naval Research Logistics Quarterly, 22(1975), 297-315.

[8] Balachandran, V. and Thompson, G.L., "An Operator Theory of Parametric Programming for the Generalized Transportation Problem-IV: Global Operators," Naval Research Logistics Quarterly, 22 (1975), 317-339.

[9] Bansal, P.P. and Johnson, S.E., "An Algorithm for Optimizing Network Flow Capacity Under Economics of Scale," Journal of Optimization Theory and Applications, 15(1975), 565-586.

[10] Bazaraa, M.S. and Jarvis, J.J., "Linear Programming and Network Flows," John Wiley and Sons, Inc., New York, 1977.

[11] Beale, E.M.L., "An Algorithm for Solving the Transportation Problems when the Shipping Cost Over Each Route is Convex," Naval Research Logistic Quarterly, 6 (1959), 43-56.

[12] Berman, E.B., "Resource Allocation in a PERT Network Under Continuous Activity Time-Cost Functions," Management Science, 10(1964), 734-745.

[13] Cabot, V., Francis, R.L. and Stary, M.A., "A Network Flow Solution to a Rectilinear Distance Facility Location Problem" AIIE Transactions, 26 (1970), 132-141.

[14] Chandrasekaran, R., "Minimum Ratio Spanning Trees," Networks, 7 (1977), 335-342.

[15] Charnes, A. and Cooper, W.W., "Management Models and Industrial Applications of Linear Programming," Vol. I and II, John Wiley and Sons, Inc., New York, 1961.

[16] Charnes, A. and Klingman, D., "The Distribution Problem with Upper and Lower Bounds on the Node Requirements," Management Science, 16 (1970), 638-642.

[17] Charnes, A., Glover, F. and Klingman, D., "The Lower Bounded and Partial Upper Bounded Distribution Model," Naval Research Logistics Quarterly, 18 (1971), 277-281.

[18] Chen, S. and Saigal, R., "A Primal Algorithm for Solving a Capacitated Network Flow Problem with Additional Linear Constraints," Networks, 7 (1977), 59-79.

[19] Christofides, N. and Brooker, P., "Optimal Expansion of an Existing Network," Mathematical Programming, 6 (1974), 197-211.

[20] Clarke, C.E., "The Optimum Allocation of Resources Among the Activities of a Network," Journal of Industrial Engineering, 12(1961), 11-17.

[21] Dantzig, G.B., "Linear Programming Under Uncertainty," Management Science, 1, (1955), 197-206.

[22] Dantzig, G.B., "Linear Programming and Extensions," Princeton University Press, Princeton, N.J., 1963.

[23] Dearing, P.M. and Francis, R.L., "A Network Flow Solution to a Multifacility Minimax Location Problem Involving Rectilinear Distances," Transportation Science, 8 (1974), 126-141.

[24] Dinkelback, W., "On Nonlinear Fractional Programming," Management Science, 13 (1967), 492-498.

[25] Dorsey, R.C., Hodgson, T.J. and Ratliff, H.D., "A Network Approach to a Multi-Facility, Multi-Product Production Scheduling Problem Without Back Ordering," Management Science, 21 (1975), 813-822.

[26] Doulliez, P. and Rao, M.R., "Capacity of a Network with Increasing Demands and Arcs Subject to Failure," Operations Research, 19 (1971), 905-915.

[27] Douliez, P. and Rao, M.R., "Maximal Flow in a Multiterminal Network with Any one Arc Subject to Failure," Management Science, 18 (1971), 48-58.

[28] Doulliez, P. and Rao, M.R., "Optimal Network Capacity Planning: A Shortest Path Scheme," Operations Research, 23 (1975), 810-818.

[29] Eisemann, K., "The Generalized Stepping Stone Method for the Machine Loading Problem," Management Science, 11 (1964), 154-176.

[30] Elmaghraby, S.E., "Sensitivity Analysis of Multiterminal Flow Networks," Operations Research, 12 (1964), 680-688.

[31] Elmaghraby, S.E., "The Determination of Optimal Activity Duration in Project Scheduling," Journal of Industrial Engineering, 19 (1968), 48-51.

[32] Evans, J.R., "Some Network Flow Models and Heuristics for Multiproduct Production and Inventory Planning," AIIE Transactions, 9 (1977), 75-81.

[33] Fair, G.M. and Geyer, J.C., "Water Supply and Waste Water Disposal," John Wiley and Sons, Inc., New York, 1954.

[34] Ferguson, A.R. and Dantzig, G.B., "The Allocation of Aircraft to Routes-An Example of Linear Programming Under Uncertain Demands," Management Science, 3 (1956), 45-73.

[35] Fillet, A., Giraud, P. and Sakarovitch, A., "An Algorithm to Find Minimum Cost Flow in Networks," Research Report ORC70-37, University of California, Berkley, 1970.

[36] Florian, M. and Roberts, P., "A Direct Method to Locate Negative Cycles in a Graph," Management Science, 17 (1971), 307-310.

[37] Ford, L.R. and Fulkerson, D.R., "Maximal Flow Through a Network," Canadian Journal of Mathematics, 8 (1956), 399-404.

[38] Ford, L.R. and Fulkerson, D.R., "Flows in Networks," Princeton University Press, Princeton, N.J., 1962.

[39] Fulkerson, D.R., "Increasing the Capacity of a Network: The Parametric Budget Problem," Management Science, 5(1959), 472-483.

[40] Fulkerson, D.R., "A Network Flow Computation for Project Cost Curve," Management Science, 7 (1961), 167-178.

[41] Gass, S.I. and Saaty, T.L., "The Computational Algorithm for the Parametric Objective Function," Naval Research Logistics Quarterly, 2 (1955), 39-55.

[42] Gass, S.I. and Saaty, T.L., "Parametric Objective Function. Part II: Generalizations," Operations Research, 3 (1955), 395-401.

[43] Glover, F., Karney, D., Klingman, D. and Russell, R., "Solving Singly Constrained Transshipment Problems," Transportation Science, 12 (1978), 277-297.

[44] Glover, F. and Klingman, D., "Network Formulations of Cash Flow Problems," Research Report, Center of Cybernetics Studies, University of Texas, Austin, 1975.

[45] Gomory, R.E. and Hu, T.C., "Synthesis of a Communication Network," SIAM Journal of Applied Mathematics, 12(1964), 348-369.

[46] Goyal, S.K., "A Note on a Simple Time-Cost Trade-off Algorithm," Management Science, 21 (1975), 718-722.

[47] Grinold, R.C., "Calculating Maximal Flow in a Network with Positive Gains," Operations Research, 21 (1973), 528-541.

[48] Hammer, P.L., "Increasing the Capacity of a Networks," CORS Journal, 7 (1969), 83-91.

[49] Handler, G.Y. and Zang, I., "A Dual Algorithm for the Constrained Shortest Path Problem," Networks, 10 (1980), 293-310.

[50] Hess, M.L., "An Observation on the Branch and Bound Method of Price," CORS Journal, 6 (1968), 58.

[51] Hu, T.C., "Minimum Cost Flow in Convex Cost Networks," Naval Research Logistics Quarterly, 13 (1966), 1-9.

[52] Hu, T.C., "Integer Programming and Network Flows," Addison Wesley, Reading, Mass., 1969.

[53] Hultz, J. and Klingman, D., "Solving Constrained Generalized Network Problems," Research Report CCS 257, University of Texas, Austin, 1976.

[54] Hultz, J. and Klingman, D., "Solving Singularly Constrained Network Problems," Applied Mathematics and Optimization, 4 (1978), 103-119.

[55] Jagannathan, R., "On Some Properties of Programming Problems in Parametric Form Pertaining to Fractional Programming," Management Science, 12 (1966), 863-888.

[56] Johnson, E.L., "Networks and Basic Solutions," Operations Research, 14 (1966), 619-623.

[57] Joksch, H.C., "The Shortest Path Problem with Constraints," Journal of Mathematical Analysis and Applications, 14 (1966), 191-197.

[58] Kelley, J.R., Jr., "Critical Path Planning and Scheduling: Mathematical Basis," Operations Research, 9(1961), 296-320.

[59] Klein, M., "A Primal Method for Minimum Cost Flows," Management Science, 14 (1976), 205-220.

[60] Klingman, D. and Ross, G.T., "Finding Equivalent Network Formulations for Constrained Network Problems," Management Science, 23 (1977), 737-744.

[61] Klingman, D. and Russell, R., "On Solving Constrained Transportation Problems," Operations Research, 23 (1975), 91-106.

[62] Klingman, D. and Russell, D., "A Streamlined Simplex Approach to the Singly Constrained Transportation Problems," Naval Research Logistics Quarterly, 25 (1978), 681-695.

[63] Lamberson, L.R. and Hocking, R.R., "Optimum Time Compression in Project Scheduling," Management Science, 16 (1970), B597-B606.

[64] Lawler, E.L., "Combinatorial Optimization: Networks and Matroids," Holt, Rinehart and Winston, 1976.

[65] Masch, V.V., "The Cyclic Method for Solving the Transshipment Problem with an Additional Linear Constraint," Networks, 10(1980), 17-32.

[66] McWhite, P.B. and Ratliff, H.D., "Modelling Mining Attrition Using Networks with Gains," Research Report 73-4, Industrial and Systems Engg. Department, University of Florida, Gainesville, 1973.

[67] Menon, V.V., "The Minimum Cost Flow Problem with Convex Costs," Naval Research Logistics Quarterly, 12 (1965), 163-172.

[68] Minieka, E., "Parametric Network Flows," Operations Research, 20(1972), 1162-1167.

[69] Minieka, E., "A Note on Parametric Network Flows," Management Science, 19 (1973), 585-587.

[70] Minty, G., "Monotone Networks," Proceedings of the Royal Society of London," Series A, 257 (1960), 194-212.

[71] Minty, G., "Solving Steady State Nonlinear Networks of Monotone Elements," Transactions P.G.C.T. (IRE), CT-8 (1961), 99-104.

[72] Murty, K.G., "Linear and Combinatorial Programming," John Wiley and Sons, Inc., New York, 1976.

[73] Oettli, W. and Prager, W., "Optimal and Suboptimal Capacity Allocation in Communication Networks," Journal of Optimization Theory and Applications, 8(1971), 396-411.

[74] Onaga, K., "Optimal Flows in General Communication Networks," Journal of Franklin Institute, 283 (1967), 308-327.

[75] Panwalkar, S.S., "Parametric Analysis of Linear Programs with Upper Bounded Variables," Naval Research Logistics Quarterly, 20 (1973), 83-93.

[76] Picard, J.C. and Ratliff, H.D., "A Cut Approach to the Rectilinear Distance Facility Location Problem," Operations Research, 26 (1978), 422-433.

[77] Picard, J.C. and Ratliff, H.D., "A Cut Approach to a Class of Quadratic Integer Programming Problems," Networks, 10 (1980), 363-370.

[78] Phillips, S. and Dessouky, M.I., "Solving the Project Time-Cost Tradeoff Problems Using the Minimal Cut Concept," Management Science, 24 (1977), 393-400.

[79] Price, W.L., "Increasing the Capacity of a Network where the Costs are Nonlinear: A Branch and Bound Approach," CORS Journal, 5 (1967), 110-114.

[80] Robinson, D.R., "A Dynamic Programming Solution to Cost-Time Trade-off for CPM," Management Science, 22 (1975), 158-166.

[81] Saaty, T.L. and Gass, S.I., "The Parametric Objective Function, Part I," Operations Research, 2 (1954), 316-319.

[82] Segal, M., "The Operator-Scheduling Problem: A Network-Flow Approach," Operations Research, 22 (1974), 808-823.

[83] Shier, D.R. and Witzgall, C., "Arc Tolerances in Shortest Path and Network Flow Problems," Networks, 10 (1980), 277-292.

[84] Siemens, N., "A Simple CPM Time-Cost Trade-off Algorithm," Management Science, 17 (1971), B354-B363.

[85] Srinivasan, V. and Thompson, G.L., "An Operator Theory of Parametric Programming for the Transportation Problem-I," Naval Research Logistics Quarterly, 19 (1972), 205-225.

[86] Srinivasan, V. and Thompson, G.L., "An Operator Theory of Parametric Programming for the Transportation Problem-II," Naval Research Logistics Quarterly, 19 (1972), 227-252.

[87] Srinivasan, V. and Thompson, G.L., "Solving Scheduling Problem by Applying Cost Operators to Assignment Models," in Symposium on the Theory of Scheduling and its Applications, Edited by S.E. Elmaghraby, Springer Verlag, New York, 1973.

[88] Srinivasan, V., "A Transshipment Model for Cash Management Decisions," Management Science, 20 (1974), 1350-1363.

[89] Tomlin, J.A., "Branch and Bound Methods for Integer and Non-Convex Programming," in Integer and Nonlinear Programming, Edited by J. Abadie, North Holland Publishing Company, Amsterdam, Holland, 1970.

[90] Truemper, K., "On Max Flow with Gains and Pure Min-Cost Flows," SIAM Journal of Applied Mathematics, 32 (1977), 450-456.

[91] Veinott, A.F., "Least d-Majorized Network Flows with Inventory and Statistical Applications," Management Science, 9 (1971), 547-567.

[92] Wagner, H.M., "Principles of Operations Research," Prentice-Hall, Inc., Englewood Cliffs, N.J., 1967.

[93] Weintraub, A.F., "A Primal Algorithm to Solve Network Flow Problem with Convex Costs," Management Science, 21 (1974), 87-97.

[94] Wollmer, R.D., "Stochastic Sensitivity Analysis of Maximum Flow and Shortest Route Networks," Management Science, 14(1968), 551-564.

[95] Zangwill, W., "A Backlogging Model and a Multi-Echelon Model of a Dynamic Economic lot Size Production System-A Network Approach," Management Science, 15 (1969), 506-527.

[96] Klingman, D., Napier, A. and Stutz, J., "NETGEN-A Program for Generating Large Scale (Un) Capacitated Assignment, Transportation and Minimum Cost Flow Network Problems," Management Science, 20 (1974), 814-821.